A real-time algorithm for acoustic monitoring of ingestive behavior of grazing cattle

José O. Chelotti[a], Sebastián R. Vanrell[a], Diego H. Milone[a], Santiago A. Utsumi[b], Julio R. Galli[c], H. Leonardo Rufiner[a,d], Leonardo L. Giovanini[a]

[a] *Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, sinc(i), FICH-UNL/CONICET, Argentina*
[b] *W.K. Kellogg Biological Station and Departament of Animal Science, Michigan State University, United States*
[c] *Facultad de Ciencias Agrarias, Universidad Nacional de Rosario, Argentina*
[d] *Facultad de Ingeniería, Universidad Nacional de Entre Ríos, Argentina*

Abstract

Assessment of both grazing behavior and herbage intake are two very difficult tasks that can be concurrently accomplished by means of accurate detection, classification and measurement of grazing events such as chews, bites and chew-bites. It is well known that acoustic monitoring is among the best methods to automatically quantify and classify ingestive and rumination events in grazing animals. However, most existing methods of signal analysis appear to be computationally complex and costly, and are therefore difficult to implement. In this work, we present and test a novel analysis system called Chew-Bite Real-Time Algorithm (CBRTA) that works fully automatically in real-time to detect and classify ingestive events of grazing cattle. The system employs a directional wide-frequency microphone facing inwards on the forehead of animals, and a coupled signal analysis and de-

Email address: `jchelotti@sinc.unl.edu.ar` (José O. Chelotti)

cision logic algorithm that measures shape, amplitude, duration and energy of sound signals to iteratively detect and classify ingestive events. Performance and validation of the CBRTA was determined using two databases of grazing signals. Signals were recorded on dairy cows offered either, natural pasture ($N = 25$), or experimental micro-swards in indoor controlled environment ($N = 50$). The CBRTA exhibited a simple linear complexity capable to execute 50 times faster than real-time and without undermining overall recognition rate and accuracy when signals were processed at 4 kHz sampling frequency and 8 bits quantization. Furthermore, CBRTA was capable to detect ingestive events with a 97.4% success rate, while achieving up to 84.0% success for their classification as exclusive chews, bites or composite chew-bites. The methodology proposed with CBRTA has promising application in embedded microcomputer systems that necessarily depend on fast real-time execution to minimize computational load, power source and storage memory. Such a system can readily facilitate the transmission of processed data through wireless network or the storage in an onboard device.

Keywords: Acoustic monitoring, cattle grazing behavior, jaw movement classification, real-time execution, signal processing.

1. Introduction

Accurate monitoring of livestock grazing behavior is necessary to ensure that most basic requirements of animal health and welfare are met and consistent with practices that can assure sustainable and efficient use of grazing resources. Hence, different efforts have been put into finding most appropriate techniques to measure and monitor diet and feeding behavior of free-grazing animals (Hodgson et al., 1996; Delagarde et al., 1999). One possible and reliable way is through the detection of distinct jaw movements associated with three common basic events: bites, chews and compound chew-bites (Milone et al., 2012). A grazing bite includes the apprehension and severance of herbage, while a grazing or rumination chew includes the crushing, grinding and processing of consumed herbage. The chew-bite is a third important grazing event that results from the overlapping of chewing and biting on a same jaw movement. Thus, jaw movements can serve as a reliable measure of distinct grazing and rumination cycles. Furthermore, the quantification of rumination chews could provide rich information on the ruminal fermentation of fiber and correlated changes in rumen pH (Sauvant, 2000). Likewise, herbage intake rate appears to depend on trade-offs between ingestive bites, chews and chew-bites, and the monitoring of these events could therefore inform on the ability of grazing herbivores to modulate changes in intake rate (Laca et al., 2000). While the number and characteristics of grazing and rumination events vary according to several plant, animal and environmental factors, they could be monitored as indicators of animal health, welfare or nutritional status (De Boever et al., 1990). To the best of our knowledge, only few studies have been focused on developing automated systems to monitoring changes in grazing and rumination.

One plausible approach to measure feeding behavior is acoustic monitoring. Alkon and Cohen (1986) and Alkon et al. (1989) used acoustic biotelemetry to study the feeding behavior of porcupine. Laca et al. (1992) instrumented an inward-facing microphone on the forehead of steers to register stronger and readily distinguishable sounds of bites, chews and chew-bites.

Consequently, acoustic monitoring proved to be a more effective methodology to discriminate sensitive differences in feeding and rumination than previous jaw recorders or visual observation methods (Ungar and Rutter, 2006), and since then it has been increasingly applied as a research tool to study different aspects of grazing behavior in sheep and cattle (Galli et al., 2006, 2011).

Broad application of acoustic monitoring continues to depend on suitable algorithms for automatic recognition of sound signals associated with chewing and biting. Milone et al. (2009) used concepts of automatic speech recognition and Hidden Markov Models (HMM) to develop an algorithm for both detection and classification of chewing and biting. The algorithm successfully detected 89%, 58%, and 56% of chews, bites and compound chew-bites in grazing sheep, respectively. Galli et al. (2011) further tested this algorithm to demonstrate the feasibility of using acoustic variables to estimate herbage dry matter intake in grazing sheep. Subsequently, Milone et al. (2012) developed a new algorithm that hereafter will be referred as CBHMM (Chew-Bite Hidden Markov Model) that extended upon previous HMMs. The CBHMM was developed for both detection and classification of chews, bites and chew-bites, in grazing cattle; obtaining up to 85% successful recognition rate.

In an independent development Clapham et al. (2011) adapted the use of SIGNAL software (Engineering Design, Berkeley, CA) for analysis of grazing sounds in cattle. The software was operated on a careful calibration to detect bites in the band of 17 kHz to 22 kHz, and on a high-pass filter with cutoff frequency at 600 Hz to attenuate background noise. The software detected bites with a 95% confidence, but it seems to demand careful and site-specific calibration before it can be used with different animals, pastures or experimental conditions. The capacity of the recording device and power source were among other limitations of the proposed system.

Navon et al. (2013) implemented an algorithm that used a machine-learning approach to analyze time-domain features (i.e., shape, intensity, duration and sequence of events) of ingestive sounds in grazing cattle. The procedure eliminated the need of calibrations and allowed a detection of ingestive events with a 94% correct and 7% false identification. More recently, Tani et al. (2013) applied pattern recognition techniques to iteratively measure eating and ruminating events collected by a single-axis accelerometer. The recognition patterns were defined in frequency domain and used to identify and classify likely eating and rumination events. Without previous calibration, recognition results were similar to previous analytical procedures used by Clapham et al. (2011) and Navon et al. (2013) . However, likely limitations of the methodology were associated with the spectral similarities between rumination and eating signals, presence of non-stationary background noise, and high computational cost associated with the analysis of signals sampled at high frequency.

Although several of the previous instrumentation and analytical procedures have shown good performance for detection of signals associated with eating and/or rumination, few of them offered possibilities to accurately classify exclusive bites, chews and chew-bites, which is a necessary condition for reliable measures of grazing behavior and even for estimation of herbage intake by means of acoustic methods. Furthermore, most if not all of previous methodologies deal with high quality and long duration signals (hours or days) that can demand collection, recording, storage, transfer and analysis of data by means of computationally complex and costly procedures, that can quickly undermine their application as fast, efficient and timely monitoring systems.

The main objectives of the present work were: 1) to develop a novel algorithm called CBRTA (Chew-Bite Real-Time Algorithm) that can be executed in real-time for automatic and efficient identification and classification of chews, bites and chew-bites, 2) to provide an analysis of the computational complexity of CBRTA, 3) to examine the operational performance of CBRTA as a function of modifications in algorithm parameters, and, 4) to provide a validation of CBRTA for both detection and classification of ingestive events in cattle by using two databases of acoustic monitoring of dairy cows grazing either outdoor temperate pasture or micro-swards in indoor controlled environment. Outdoor grazing environments inevitably introduce some level of unpredictable and variable background noise that can readily interfere with the acquisition and analysis of chewing and biting signals. We aimed therefore to deal with commonly encountered levels of such noises by combining passive isolation (directional microphones with isolation material) and basic signal processing.

2. The algorithm

The design goal was the achievement of an algorithm that can combine high performance for detection and classification of sound events with low computational cost, which is a necessary condition to allow real-time execution of the algorithm in portable embedded systems. To achieve this goal, time-domain instead of transformed-domain (frequency, time-frequency) analysis was implemented to avoid high computational load of signal analysis.

2.1. General description

Signals associated with an exclusive chew (Figure 1a), composite chew-bite (Figure 1b) or exclusive bite (Figure 1c) have readily distinguishable properties.

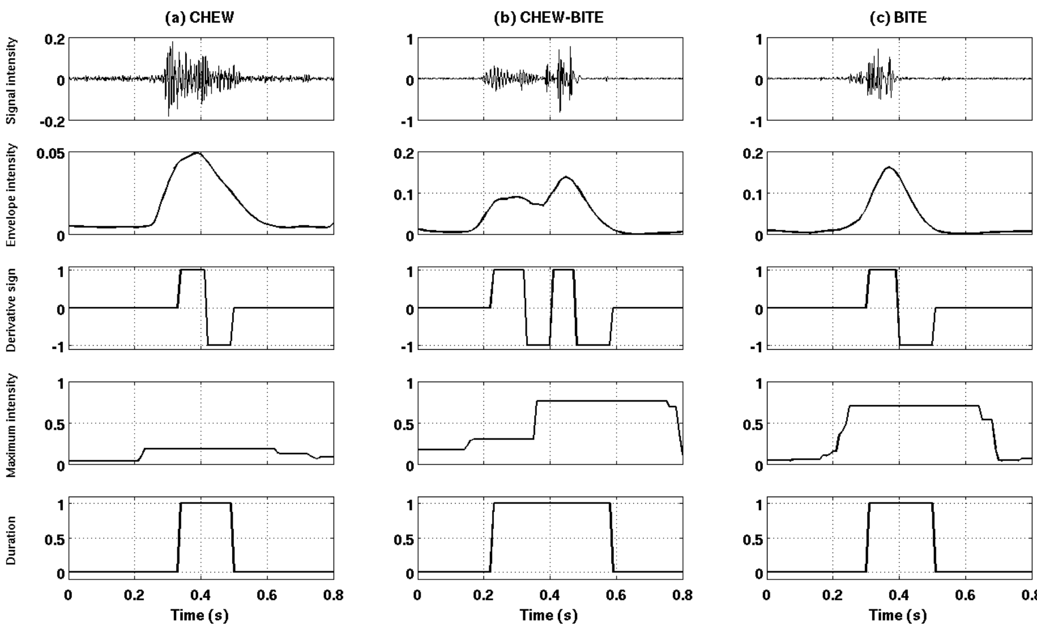

Figure 1: Examples of typical acoustic events produced by jaw movements and their correspondent features: (a) chew, (b) chew-bite and (c) bite. Within each row, top-down: raw acoustic signal, computed envelope, sign of envelope slope, maximum intensity and duration are shown.

Therefore, the shape, maximum intensity and duration of sounds were isolated to discriminate among the bites, chews and chew-bites. The shape of a jaw movement is characterized by changes in both the intensity and sign of the envelope slope (Figure 1). The sign (either positive or negative) of the envelope slope changes one or two times for chews and bites and more than two times for composite chew-bites. The three jaw movements also produce sounds with distinguishable maximum intensity that remains low for chews and high for bites, and changes from low to high for composite chew-bites. Finally, bites, chews and composite chew-bites, have a defined duration, which is shorter for chews and bites and longer for composite chew-bites (Figure 1).

Sound properties were then used by the algorithm to complete two successive tasks, event detection and event classification, respectively. For the detection task, the algorithm detects the region of the sound envelope that shows the occurrence of a possible jaw movement. This detection is carried out through the identification of characteristic peaks in the sound envelope using an adaptive threshold. For the classification task the algorithm uses a simple set of rules to compute and compare the shape, intensity and duration of a detected event to a given threshold value.

For implementation purposes, the completion of the two tasks can be thought as a set of five successive stages, where the first four stages are used to complete the event detection task, while the event classification task is performed during the last stage, as follows.

Stage 1 - Envelope computation: One basic requirement for the implementation of the algorithm is the envelope computation, which is decomposed into three steps: i) signal rectification, ii) signal filtering and iii) signal subsampling. In the first step the absolute value of signal samples is com-

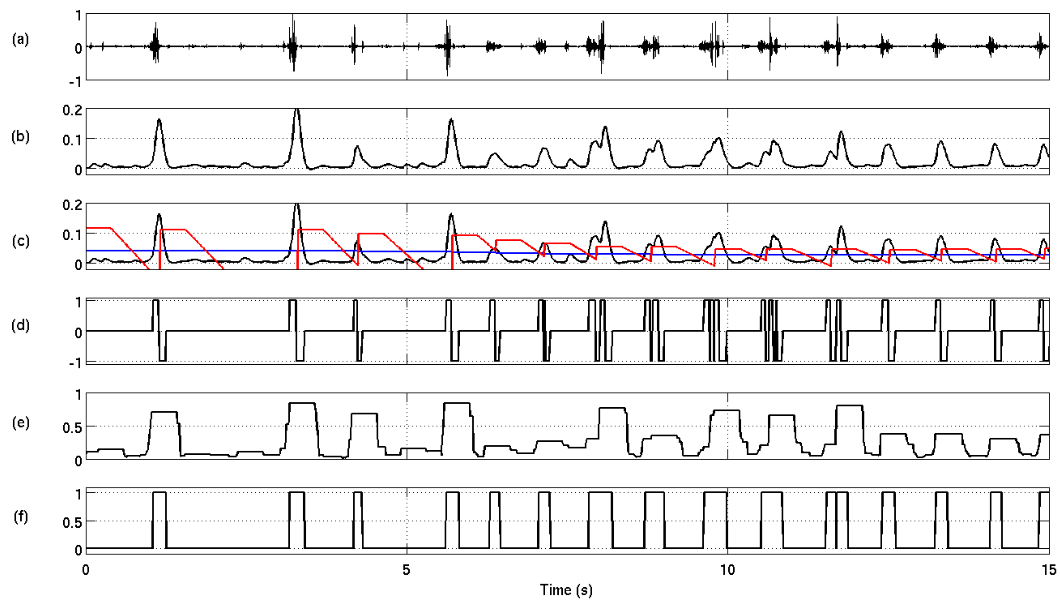

Figure 2: Example of a 15s sound track with correspondent signals generated by the processing algorithm: (a) original raw signal, (b) sound envelope computation, (c) event detection, (d) slope sign, (e) maximum amplitude, and (f) duration of detected events.

puted at the original sampling frequency. In the second step, the signal is filtered using a second-order low-pass Butterworth filter with a bandwidth of 5.5 Hz, producing the sound envelope. In the third step, a subsample of the original sound envelope to 100Hz is conducted (Figure 2b). The main objective of this task is to reduce the computational requirements (load and computation time) in the subsequent tasks, since this process significantly reduces the amount of information to be processed but without compromising accuracy in the detection and classification of sounds.

Stage 2 - Division of sound into segments: Short segments have lower computational resource constraints, are easier to handle, and their use can facilitate the treatment of unexpected events that need special attention. Such events include intense external noises of short duration and background noises. The size of segments depends on the computational resources that are available to implement the algorithm. In a common desktop computer segments can have a typical duration of 30 s or longer. In an embedded system with low computational capacity segments should have a smaller size. Ultimately, segment size depends on the amount of memory available for signal analysis (minimum size of 2 s).

Stage 3 - Event detection: The presence of peaks in the sound envelope reveals possible target events. Each peak is detected as a change in the derivative of the envelope. However, to be considered a possible event it must be higher than given thresholds. The peaks are detected through the comparison of the sound envelope with a time-varying threshold $T(k)$ (red dashed line in Figure 2c), where k is a time variable. This threshold is generated by an algorithm that considers both anatomical and behavioral characteristics of the animal according to the following two rules: i) a minimum period of time between two consecutive jaw movements, and, ii) a maximum duration of jaw movements within a continuous activity (i.e. ruminating or grazing). Then, following Christov (2004) the event detection algorithm uses this criteria to generate the time-varying threshold $T(k)$ with the following features:

- **Unresponsive period** (T_U): period of time after detecting an event in which the algorithm is no longer searching for a new event. It is computed for each event as a fraction α ($0 < \alpha < 1$) of the average duration of the last five events detected.

- **Maximum period** (T_M): maximum time that an event can last within the same activity. It is computed for each event as β ($\beta \geq 1$) times the average duration of the last five events detected.

- **Peak expectation threshold** (T_P): minimum value expected for the next peak intensity (blue dot-dash line in Figure 2c). It is computed as a fraction γ ($0 < \gamma \leq 1$) of the moving average of the last five peaks detected in the envelope signal

$$T_P(k) = \frac{\gamma}{5} \sum_{i=1}^{5} S_P(j-i)). \tag{1}$$

  where S_P is the peak intensity of an event, and j is an event counter.

- **Threshold slew-rate** (ΔT): is the decrease of threshold $T(k)$ once after the unresponsive period T_U expires, and serves therefore to significantly improve the event detection sensitivity. The threshold $T(k)$ only changes during the time period between T_U and T_M, as follows

$$T(k) = T(k-1) - \Delta T \;, \quad \forall \;\; T_U < k < T_M. \tag{2}$$

This stage of the algorithm generates a temporary file with correspondent timestamps to indicate the location of all detected peaks. This peak reference is then used in subsequent event detection and classification stages to trigger the analysis of signal properties.

Stage 4 - Properties computation: This step computes the shape, maximum intensity and duration of the sound to classify likely candidate events detected in previous stages. The shape of the event is computed as the number of changes (NC) in the sign of the envelope slope (Figure 2d).

To avoid confusion with noises, the slope is computed only if the magnitude of the sound envelope is bigger than the background noise (NT) detected in the analyzed segment. The maximum intensity of the envelope sound (EA) is computed directly from the absolute value of the signal over a window of time whose length is half of the duration of a typical chew-bite event (Figure 2e). The duration of the event (ED) is determined from the sound envelope by measuring the time period when the sound envelope is bigger than the background noise NT (Figure 2f).

Stage 5 - Event classification: Using a specific set of rules, based on previously computed properties, each event is classified into one of five categories: chew (C), bite (B), chew-bite (CB), silence (S) or noise (N). Briefly, the algorithm explores the timestamp, NC, EA and ED to detect and classify the events. The algorithm applies a set of rules to find whether a true event has happened or not and, in a positive case, which kind of event has been detected. The set of rules employed by the algorithm are established heuristically from a training data set, under the constraints that the set of rules should be small. The set of decision rules is detailed in Table 1. Each rule specifies the conditions that NC, EA and ED must meet to be classified as C, B or CB, respectively. For example, if NC is greater than 2, EA exceeds NT and ED is greater than 0.3 s, then the detected event is classified as CB.

Figure 3 shows the flow diagram of the algorithm, integrating all steps for envelope computation, segmentation, detection and classification of jaw movements. The envelope signal $Sp(k)$ is loaded and analyzed by segments of N samples. When a segment is fully analyzed, the results are saved before analyzing the next segment. In the first stage, the algorithm computes the

Table 1: Rules for jaw movement event classification† .

Event	Rule
Chew-bite	**if** $NC > 2$ **and** $EA > NT$ **and** $ED > 0.3[s]$ **then** $L(j)$=CB
Bite	**if** $NC <= 2$ **and** $EA >= 0.5\ T_P$ **and** $ED < 0.3[s]$ **then** $L(j)$=B
Chew	**if** $NC <= 2$ $EA > NT$ **and** $EA < 0.5T_P$ **and** $ED < 0.3[s]$ **then** $L(j)$=C

† NC is the number of changes in the sign of the slope of sound envelope, EA is the maximum intensity of the envelope, ED is the duration of the event, NT is the background noise threshold and T_P is the peak expectation threshold.

time-varying threshold $T(k)$. Then, it checks if a peak has been detected. If no peaks have been detected, the algorithm assigns the silence label (S) to the event. If a peak has been detected, the algorithm classifies the event by applying rules based on the event properties NC, EA and ED, and by assigning the correspondent label C, B, CB or N.

3. Materials and methods

Acoustic monitoring of grazing dairy cattle was used to test the performance of the algorithm and its software implementation. Signals were recorded on a different duration (in some cases several hours) but for analysis and testing only maximum periods of 5 minutes were considered, given the practical difficulty of labeling aurally longer periods. It was also necessary to establish performance measures for analysis purposes.

3.1. Experimental field conditions for collection of datasets

Two databases were obtained under different grazing conditions, and at different times and locations. The first database included signals of dairy

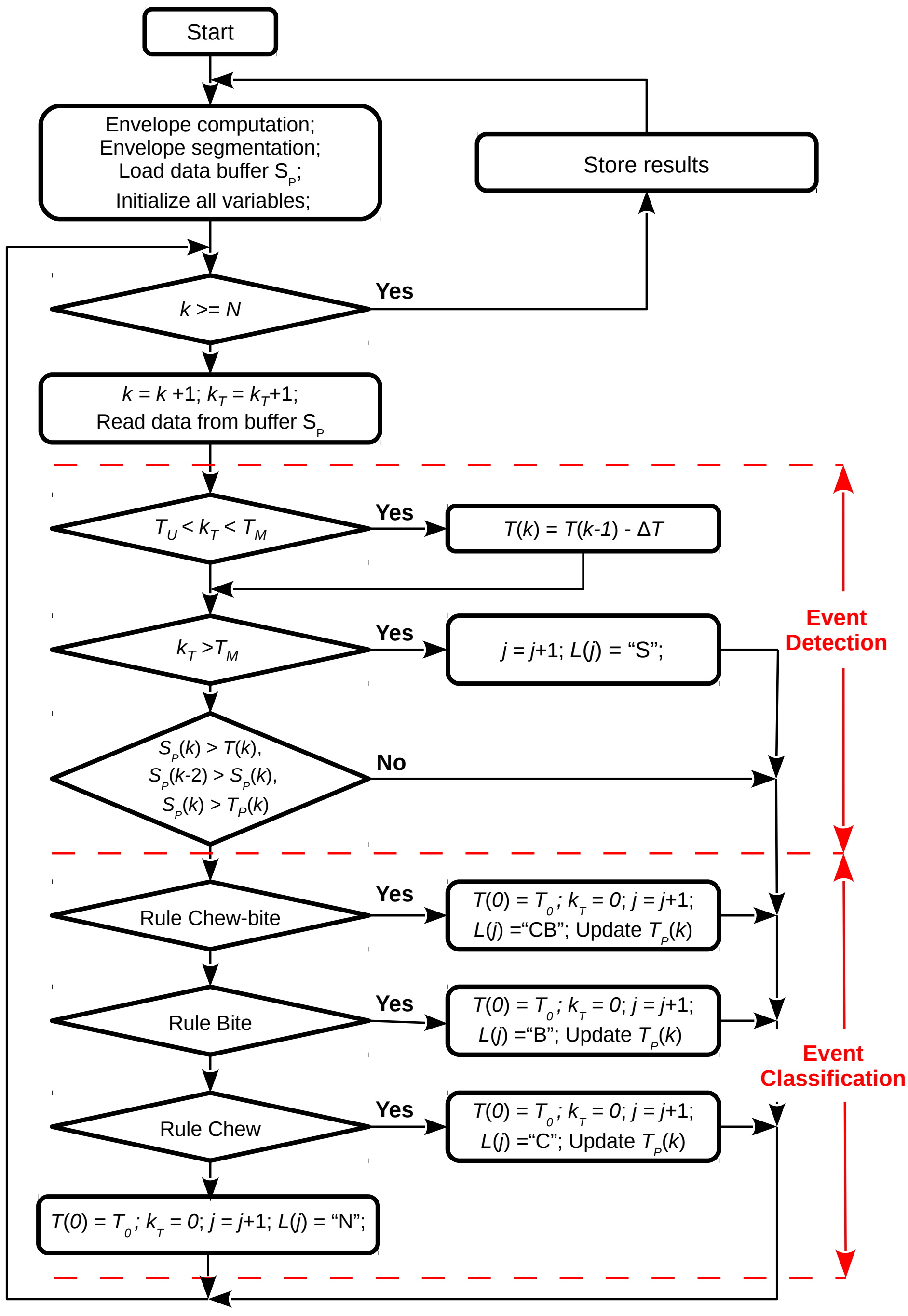

Figure 3: Flow diagram of the algorithm for event detection and classification.

cows grazing temperate pasture and was therefore useful to test the algorithm in an outdoor noisy environment. Signals for the second database were collected with dairy cows grazing micro-swards in an indoor controlled experiment. This database was used to further analyze the effect of forages (species and height) on the detection and classification capabilities of the algorithm, and to compare the performance of the algorithm against the previous CBHMM methodology developed by Milone et al. (2012).

The first database was obtained from an experiment performed at the W.K. Kellogg Biological Station dairy facility of Michigan State University, Hickory Corners, USA, during August of 2014. Protocols for animal handling and care were reviewed, approved and conducted according to the Institutional Animal Care and Use Committee of Michigan State University. In this experiment the daily foraging behavior of five multiparous lactating Holstein cows grazing perennial ryegrass/white clover and orchardgrass/white clover dominated pastures were monitored for six days, according to 5 x 5 Latin-square design to control for recording device and cow. This design therefore produced a total of 25 sound tracks of 24h duration. Cows were managed on a robotic milking system with voluntary grazing of pasture using same management protocols described in Watt et al. (2015). These signals were recorded using a SONY ICDPX312 recorder mounted on a cow halter and a directional microphone pressed onto the forehead of the cow. All recordings were made at 44.1 kHz sampling rate and 16-bit resolution, providing a nominal 22 kHz recording bandwidth and 96 dB dynamic range, and stored in the WAV (Waveform Audio) file format. Hereafter, these recordings will be referred as the Michigan Database (MDb).

The second database was the same as used by Milone et al. (2012) for development and testing of the algorithm CBHMM. Briefly, the fieldwork to obtain this database was performed at the Campo Experimental J.F. Villarino, Facultad de Ciencias Agrarias, Universidad Nacional de Rosario, Zavalla, Argentina during February 2004. Project protocols were previously evaluated and approved by the Committee on Ethical Use of Animals for Research of the Universidad Nacional de Rosario. Sound signals from dairy cows grazing either pure alfalfa or pure fescue micro-swards at two heights (tall, 24.5 ± 3.8 cm, or short, 11.6 ± 1.9 cm) were recorded individually in grazing sessions conducted over a 5-day period. Forage species were selected because they differ in sward structure and neutral detergent fiber content (alfalfa, 360 ± 11 g/kg and fescue, 631 ± 6 g/kg), which are factors that have direct influence on chewing sounds (Duizer, 2001). Two 4–6 year-old lactating Holstein cows weighing 608 ± 24.9 kg, previously tamed and trained, were used. A wireless microphone (Nady 151 VR, Nady Systems, Oakland, CA, USA) was randomly assigned to animals each day. The microphone was placed facing inwards on the forehead and was protected by rubber foam (Milone et al., 2009). The distance between the wireless microphone and the receiver was 2–3 m. Micro-swards were hand-constructed using plants in pots that were firmly attached to a baseboard placed inside a barn. Behavior was recorded with an analog video camcorder (Sony CCD-TR517), and then coded in MPG format at 25 frames per second. The sound from the wireless microphone was recorded on the tape soundtrack (16 bits, 44.1 kHz). A total of 50 grazing sessions were recorded: 15 from tall alfalfa, 11 from short alfalfa, 12 from tall fescue and 12 from short fescue. On average, for each pasture/height the signals contained approximately 13 min of recording and around 800 events (13% bites, 64% chews and 23% chew-bites). Hereafter, these recordings will be referred as the Rosario Database (RDb).

All signals were labeled aurally by experts in animal behavior to identify and classify individual events (C, B, CB, S, N) during grazing. The labeling process was done by one expert, and the result was checked by another expert. In most of the cases experts largely agreed with the labeling of signals, but when there was disagreement, both experts worked together to provide a final decision. This labeling was used as control reference for comparison and testing of the performance of the algorithm. In the case of signals belonging to MDb, two periods of 5 minutes were extracted and labeled from each 24h sound track. The signals were randomly selected within a grazing period, because during this activity the three types of events considered can be found. Each period contained approximately 350 events (25% bites, 48% chews and 27% chew-bites). One of the periods was used to analyze the effect of parameters while the other one was used for evaluation purposes. A similar data partition was made for signals belonging to RDb. For each grazing session, 50% of signals were used to analyze the effect of parameters, while the remaining 50% was used for evaluation purposes.

3.2. Performance measures

Valid comparisons between events recognized and classified by the algorithm and their corresponding reference of aurally labeled events depends on the correct synchronization of both event sequences. To solve this problem, the HTK[1]performance analysis tool HResults was used, which is based on a dynamic programming-based string alignment procedure (Young et al., 1997).

The outputs of this tool were: i) the number of deleted events (D), which are false negatives, ii) the number of substituted events (S), which are misclassified events, iii) the number of inserted events (I), which are false positives, and iv) the total number of events (T) in the reference transcription provided by the experts. An example of these definitions is shown in Table 2: the first bite of the recognized sequence is a substitution (S) because the real event is a chew; the second chewbite is an insertion (I) because there is no event in the real sequence; and the second chew in the reference sequence has not been recognized so it is a deletion (D).

Table 2: Example of sequence alignment for performance measurement.

Reference seq.:	chewbite	chew	chew	bite	chew		bite	bite
Recognized seq.:	chewbite	bite		bite	chew	chewbite	bite	bite

Keeping these definitions in mind, the percentage of detected events is computed as follows[2]

$$\delta\% = \frac{T - D}{T}\ 100\%, \tag{3}$$

the percentage number of events correctly recognized is given by

$$C\% = \frac{T - D - S}{T}\ 100\%, \tag{4}$$

[2]http://htk.eng.cam.ac.uk/

and the accuracy is computed by

$$A\% = \frac{T - D - S - I}{T}\ 100\%. \tag{5}$$

Performance of CBRTA for recognition of C, B, CB, S or N was assessed using exploratory analysis of sensitivity. This analysis computed the correspondent recognition rate $C\%$, accuracy $A\%$ and computational time as a function of changes in the following key parameters: i) the sampling frequency, ii) quantization level, iii) cut-off frequency of the detector filter, and iv) subsampling frequency. The effectiveness of CBRTA for detection of ingestive events (C, B, CB) was determined considering false negatives in the computation, but no substitutions[3]. The effectiveness of CBRTA for classification of ingestive events (C, B, CB) was determined in two ways. For the MDb database, a cross-way validation was conducted in order to demonstrate robustness. For this comparison, the CBRTA was fitted with the best set of parameters for the MDb and RDb database, respectively. By best set of parameters we means a set of parameters that provides the highest recognition rate with the highest accuracy. Then, the classification by CBRTA fitted with the best set of parameters for MDb database [CBRTA (MDb)] was compared to the correspondent classification of CBRTA fitted with the best set parameters for the RDb database [CBRTA (RDb)]. For the RDb database, the CBRTA was compared to the CBHMM algorithm of Milone et al. (2012). This comparison was decided for two reasons. The CBHMM

[2]While this computation does not include insertions (false positives), these were quantified in the present analysis.

is the only other available method that makes a distinct classification of C, B and CB, and, the CBRTA and CBHMM are both originally fitted to the same RDb database, thus offering a direct unbiased comparison of methods. On the other hand, the application of the CBHMM method on a different database could be wrong, because the models would need to be adapted to the new recording conditions. Also, it is important to note that to train and evaluate the CBHMM, a hold-out cross-validation method was used (Duda et al., 2001), while in the present CBRTA parameterization is done using a subset of RDb data not further used for testing purposes. The CBRTA was implemented using MATLAB R2010b for evaluation purposes.

Thereafter, the analysis included the testing of i) algorithm complexity, ii) computational performance, and iii) validation of CBRTA for both the automatic detection and classification of ingestive events in grazing dairy cattle. The computational complexity was modeled for each computational task as the function of the number of samples n to be processed each second. For more exhaustive analysis, the computational cost of CBRTA was compared to the CBHMM algorithm proposed by Milone et al. (2012).

4. Results

4.1. Complexity analysis

The computational cost for each step of the CBRTA algorithm evaluated as function of the number of samples n to be processed each second

[3] This is because there are no substitutions in a detection problem, since we are only interested in whether an event has occurred or not, regardless of its type. Instead, the classification stage of events should consider substitutions.

is shown in Table 3. This analysis considered a filtering task applied as second order infinite impulse response (IIR) filter. The total number of operations per second $f_{CBRTA}(n)$ required to execute the CBRTA algorithm was $f_{CBRTA}(n) = 13n + 3700$. As shown in Table 3, only the first three tasks (i.e., rectification, filtering and subsampling) will depend on the sampling frequency of the input signal. After subsampling (Stage 1), the signal processed by the remaining tasks has a constant sample rate (100 samples/s). Therefore, the remaining tasks will be independent of the audio sample rate. For example, the computation of the envelope slope requires the subtraction of two consecutive samples for computation of its sign, which involves two operations per sample. Similarly, the classification of events involves five comparisons to check whether the predefined classification conditions are met or not. A more detailed description of the complexity analysis for the CBRTA algorithm is provided in Appendix A. This analysis shows a linear computational complexity for CBRTA. A comparative analysis of complexity on the CBHMM algorithm developed by Milone et al. (2012) is summarized in Appendix B. This analysis shows a superlinear complexity for CBHMM.

4.2. Effect of parameters on system performance

Figure 4 shows the effect of sampling frequency on the performance of the algorithm (recognition rate and accuracy) and the corresponding computational time for the MDb database. The recognition rate and accuracy remained high (around 80%) over wide range of frequencies (from 2 kHz to 11 kHz) and declined for frequencies that were outside of this range. This phenomenon can be explained by the fact that for sampling frequencies below 2 kHz the signal/noise ratio is degraded because important components

Table 3: Number of operations per second of the CBRTA algorithm for detection and classification of jaw movement events.

Stage	Task	Operations/s
1	Signal rectification	$2n$
1	Signal filtering	$9n$
1	Signal subsampling	$2n$
2	Samples buffering	100
3	Threshold generation	900
3	Event detection	100
4	Envelope slope computation	200
4	Maximum signal	100
4	Event duration computation	200
5	Silence rule	100
5	Chew-bite rule	500
5	Bite rule	500
5	Chew rule	500
5	Noise rule	500

of the signal are filtered out. In a similar way, once the sampling frequency goes beyond 11 kHz, the amount of noise processed by the algorithm increases, further reducing and degrading the overall signal/noise relationship. However, in the range of frequencies from 2 kHz to 11 kHz, the information and noise processed by the algorithm remains unchanged, keeping the overall signal/noise relationship constant.

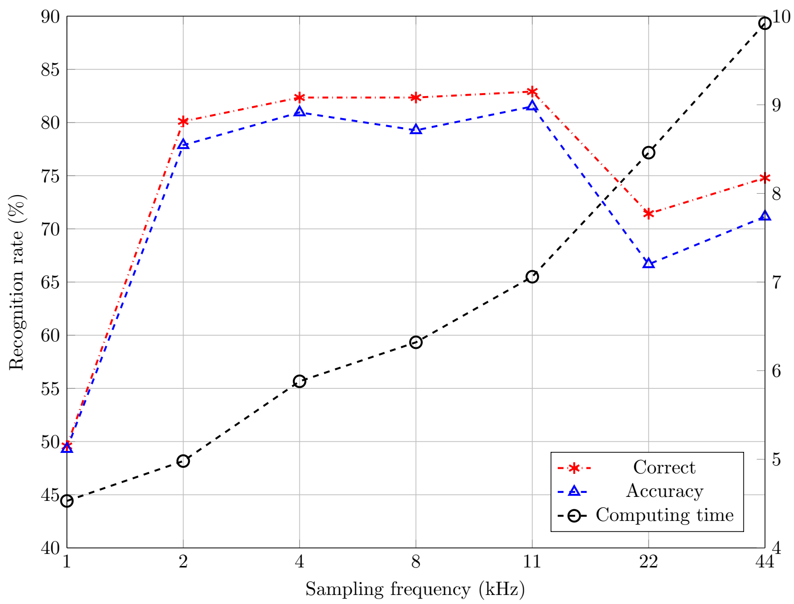

Figure 4: Algorithm recognition rate and accuracy and corresponding computational time as a function of sampling frequency for frames of 5-minute duration (N=25).

The linear dependency of the computational time with sampling frequency is shown in Figure 4. At a sampling frequency of 4 kHz the algorithm had reasonably good compromise between performance (recognition rate and accuracy) and computational time. In this sense, the algorithm proved to be capable of processing signals 50 times faster than real-time or 300 s (5 minutes) of sound signal in 6 s (Figure 4). This means that in a practical application the algorithm is capable to analyze 50 minutes of acoustic data per minute in a standard desktop computer.

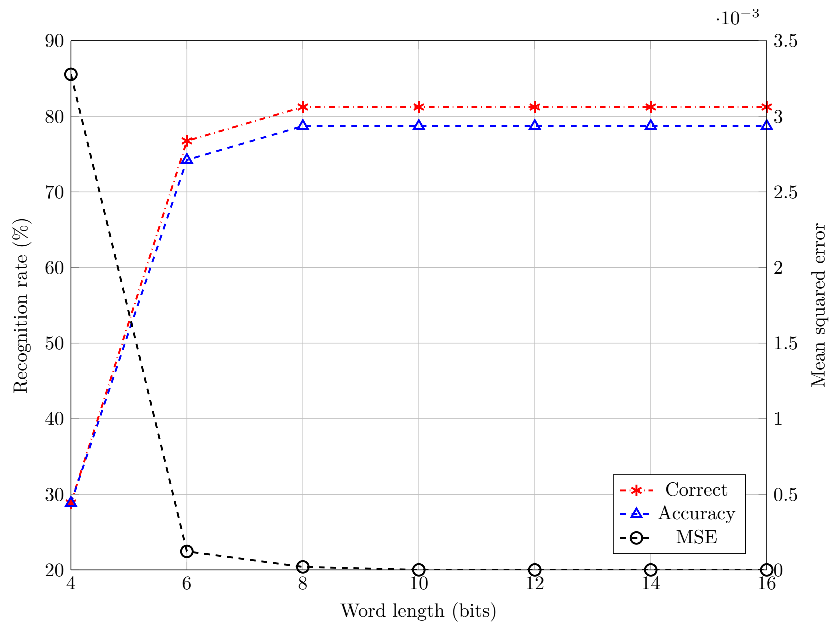

Figure 5: Algorithm recognition rate and accuracy, and corresponding computational error as a function of quantization level (computed as world length) for frames of 5-minute duration (N=25).

Figure 5 shows the effect of quantization level (or word length representation) on the performance of the algorithm (recognition rate and accuracy) for the MDb database. The recognition rate and accuracy remained high (around 80%) for a quantization level of 8 bits or more. This phenomenon can be explained by the fact that the quantization error, measured in terms of the mean square error (MSE) between the signal represented by data of a given word length (resolution) and the signal represented by data of the longest word (16 bits), is almost zero for ingestive sound data codified with

8 bits or more.

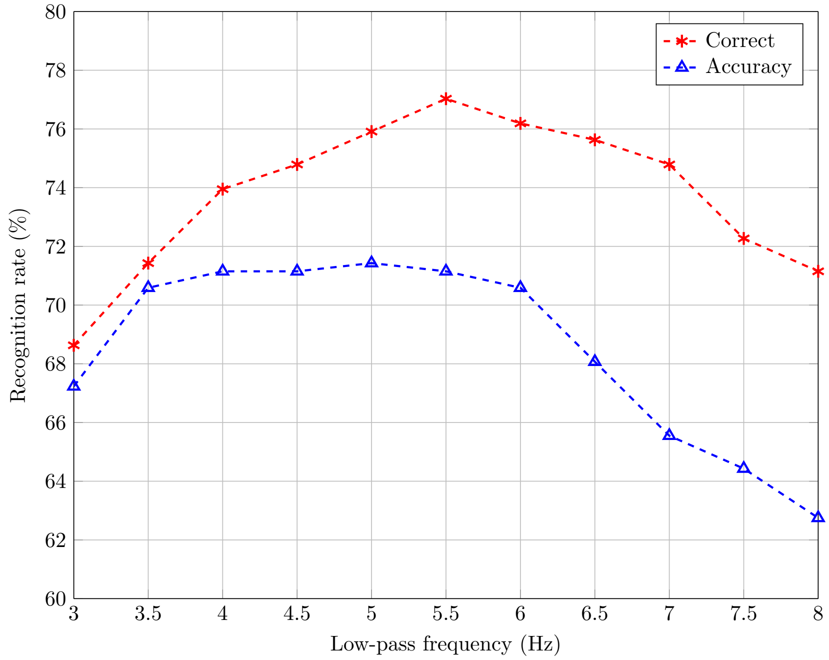

Figure 6: Algorithm recognition rate and accuracy as a function of cut-off frequency in envelope detector filter for frames of 5-minute duration (N=25).

Figure 6 shows the effect of the cut-off frequency of the envelope detector filter on the recognition rate and accuracy for the MDb database. Both, recognition rate and accuracy improved as the cut-off frequency of the filter increased from 3 Hz to 5 Hz. A correct recognition rate over 75% and accuracy over 70% was observed in the frequency range between 5 Hz and 6 Hz. Beyond 6 Hz, both recognition and accuracy declined. These phenomena can be explained by the fact that enlarging the bandwidth of the filter at low frequencies increases the amount of information processed by the algorithm, thereby augmenting the overall signal/noise relationship and the performance of the algorithm. However, once the cut-off frequency goes beyond the 6 Hz, the information remains constant, the amount of noise processed by the algorithm increases, and the overall signal/noise relationship and performance of the algorithm decreases.

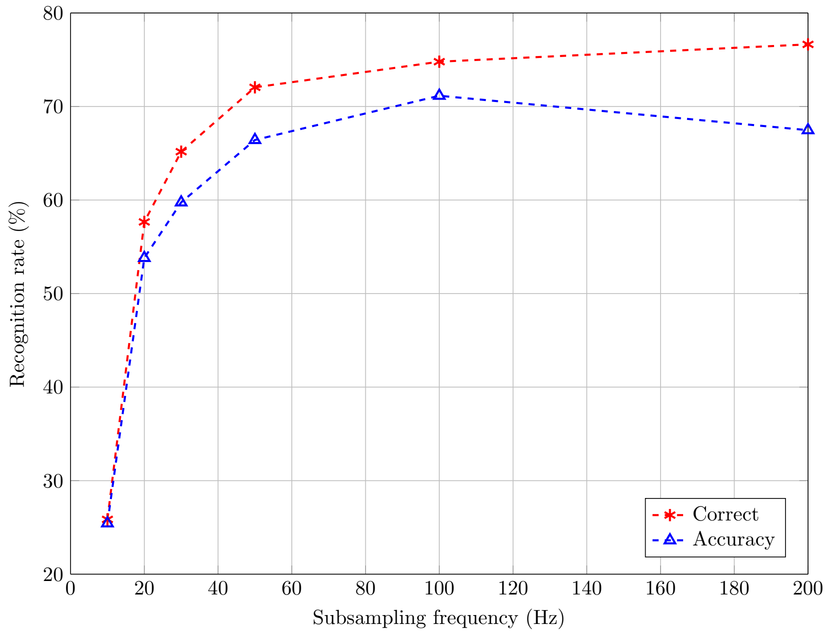

Figure 7: Algorithm recognition rate and accuracy as a function of subsampling frequency for frames of 5-minute duration.

Figure 7 shows the effect of the subsampling frequency on the algorithm recognition rate and accuracy. Both, recognition rate and accuracy were incrementally improved with increases in subsampling frequency up to 100 Hz. Beyond this subsampling frequency, the recognition rate remained steady while the accuracy showed a gradual decay. Increasing the subsampling frequency the amount information processed by the algorithm increases, improving the overall signal/noise relationship. However, once the subsampling frequency goes beyond the 100 Hz, the useful information remains constant and the overall signal/noise relationship does not suffer further change.

The algorithm performance analysis for the RDb database rendered a slightly different trend (data not shown). Best recognition rate and accuracy was observed at sampling frequencies between 2 kHz and 4 kHz. With respect to the filter cut-off frequency, the best results were observed at 3.5 Hz, where highest recognition with high accuracy was detected. Moreover, similar recognition performance was obtained at cut-off frequencies of 4 Hz and 5 Hz, but with lowering accuracy. Regarding the subsampling frequency, the best performance was observed at 100 Hz, similarly to MDb database. Also a sampling frequency of 2 kHz rendered lower overall computational cost. The differences between the parameters of the algorithm obtained for each database are primarily due to differences between the characteristics (frequency response and steady state gain, among others) of microphones used to record the databases.

4.3. Event detection and classification

As can be seen in Figure 3, when ingestive sounds are processed, two different task can be performed: i) detect the existence of an ingestive event within the record without identifying its type, and ii) classify the ingestive event by identifying the type of event detected. Clearly, the detection task is simpler and more accurate than the classification task, since it requires fewer information.

For the algorithms considered in this paper the overall detection of ingestive events was 97.4%, because of the existence 2.6% of deletions (false negatives). Also it was observed 1.4% of insertions (false positives). Regarding event classification, the CBRTA algorithm clearly distinguished among types of jaw movements in both MDb and RDb databases. In Table 4 the classification of ingestive events for the MDb is presented. The CBRTA algorithm shows an average recognition rate of 84.0% of the total events for CBRTA (MDb) and an average recognition rate of 79% of the total events for CBRTA (RDb). Therefore, the results for event classification were lower than event detection rate by an average of 15%. For both CBRTA (MDb) and CBRTA (RDb) sets of parameters the algorithm achieved good event classification rates, demonstrating ability for scalability and generalization. Also, Table 4 summarizes the recognition rates for each different event determined for the MDb database. In this table it can be observed the high ability of the algorithm to correctly identify the chew and bite events, regardless of the set of parameters used. However, some degree of confusion between bites and chew-bites was detected for the classification of chew-bites, which may be due to the close similarity of sound properties between both events. We believe that this confusion is less critical at a practical level since B and CB are both ingestive events.

Table 5 summarizes the recognition rates for different events for the RDb database. The classification results of CBRTA for this database showed an average correct classification rate of 77% of events across all pasture types, while the CBHMM method reached an average of 79% over all pastures. The best results were seen for tall pastures reaching 79% and 78% for alfalfa and fescue respectively, while for short fescue a 77% was obtained. An additional deterioration of 5% in the recognition rate can be appreciated for short alfalfa.

Table 4: Percentage of correct and false classification of bites, chews and chew-bites of dairy cows detected by a novel Chew-Bite Real-Time Algorithm (CBRTA) trained and parameterized with a same (MDb) or different (RDb) database†. Bold numbers indicate the best results.

Event	Bite		Chew		Chew-bite		Average	
	CBRTA (MDb)	CBRTA (RDb)	CBRTA (MDb)	CBRTA (RDb)	CBRTA (MDb)	CBRTA (RDb)	CBRTA (MDb)	CBRTA (RDb)
Bite	**95**	94	2	3	3	3		
Chew	8	8	87	**91**	5	1	**84**	79
Chew-bite	22	44	8	6	**70**	50		

† Testing database included acoustic records of 5 dairy cows grazing temperate pasture (N=25).

5. Discussion

Most of previous studies of acoustic monitoring in grazing ruminants were focused on the detection of ingestive or rumination events and not in their classification. To the best of our knowledge, the only algorithm that previously focused in both the automatic detection and classification of acoustic grazing events is the CBHMM method developed by Milone et al. (2012). In addition, none of the previous studies made an analysis of the computational complexity of the proposed methodologies. The computational complexity of the algorithm can impose severe limitations for implementation in a system running in real-time, and this issue becomes relevant when high quality and long duration (several hours) audio need to be processed. For real-time operation, the algorithm must be able to process a given signal segment before another segment becomes available. To accomplish this objective, the algorithm must complete at least $f(n)$ fix-point operations per second. For the

Table 5: Percentage of correct and false classification of bites, chews and chew-bites of dairy cows grazing contrasting micro-swards, detected by a former Chew-Bite Hidden Markov Model (CBHMM) or a novel Chew-Bite Real-Time Algorithm (CBRTA)†. Bold numbers indicate the best results.

		Bite		Chew		Chew-bite		Average	
		CBHMM	CBRTA	CBHMM	CBRTA	CBHMM	CBRTA	CBHMM	CBRTA
Tall alfalfa	C	**79**	67	11	18	9	15		
	B	3	2	88	**90**	9	8	**84**	79
	CB	2	5	3	11	**94**	84		
Short alfalfa	C	**76**	62	16	30	8	8		
	B	5	0	90	**94**	5	6	65	**74**
	CB	23	5	15	29	61	**66**		
Tall fescue	C	**83**	74	0	21	17	5		
	B	1	1	93	**95**	7	4	**85**	78
	CB	1	10	4	33	**94**	57		
Short fescue	C	**90**	79	9	14	1	7		
	B	0	1	**99**	**99**	1	0	**84**	77
	CB	2	25	7	32	**91**	43		

† Testing database included acoustic records of 2 dairy cows grazing a factorial set of micro-swards hand-constructed with plants in pots of 2 species (Alfalfa or Fescue) and 2 heights (short or tall), collected in 5-minute recording sessions (N = 50).

range of sampling rates considered in the CBRTA application (from 4 KHz to 44 Khz), it is easy to find a low cost commercial microprocessor capable to perform more than the required number of operations. For example, given a 44 Khz sample rate it is possible to complete the execution of CBRTA with a Tiva C microcontroller (Tiva^{TM} C Series LaunchPad Evaluation Kit, Texas Instruments Inc., Dallas, TX) using a 10 MHz clock. The processing speed could be increased further (augmenting the clock frequency), but at the expense of increasing energy consumption, which is an essential issue in portable embedded systems.

In the present study we carried out a detailed analysis of computational complexity, performance of the CBRTA algorithm and the CBHMM algorithm to then have a comparative reference of computational complexities. This analysis showed a linear computational complexity for CBRTA algorithm ($O(n)$), while for the CBHMM method was found a greater superlinear complexity ($O(n\ log(n))$). In addition to showing a lower complexity, the CBRTA algorithm had proven capability of processing grazing signals 50 times faster than real-time. Others authors such as Clapham et al. (2011) have reached up to 10 times faster than real-time but for algorithms calibrated for detection of bites alone, excluding therefore two other critical jaw events in grazing animals, exclusive chews and compound chew-bites. Thus, fast processing by CBRTA is a promising result to develop embedded microcomputer applications that depend on real-time analysis.

Indeed, a major drawback to process signals real-time on embedded systems is the computational load of the algorithm, since this can determine the requirements of hardware to implement the system. In signal processing, the computational load principally depends on two parameters: i) the sampling frequency and ii) the quantization level of the signal. The sampling frequency defines the information flow processed by the system per unit of time (Figure 4) and it plays a key role on the computational load of the algorithm (Table 3). The quantization level of the signal defines accuracy of the signal representation and, therefore the word length required by the system to process the information (Figure 5). In this way, quantization defines one aspect of the complexity of the system implementation. Our results showed that CBRTA is capable of achieving reasonable compromise between low computational time and high recognition rate and accuracy with a sampling frequency between 2 kHz and 4 kHz and a quantization of 8 bits. With this likely set of parameters both detection and classification of events rendered results that were similar to previous methodologies but at significantly lower computational cost and running time.

The overall performance of CBRTA on event detection was 97.4% across the two databases, which is in the same order of detection rate for algorithms published in the specialized literature. In this sense, Clapham et al. (2011) reported a successful detection of bites of 95%, while Navon et al. (2013) reported detection rates for jaw movements of 94% in a low noise environment. Milone et al. (2012) developed an algorithm extending from HMM models to detect and classify ingestive sounds of cattle (i.e. C, B and CB), reaching a successful detection rate of 94%. In a similar way, Tani et al. (2013) detected ingestive and ruminating chewing with approximately a 98% detection success. These quantitative results (except the results of algorithm developed by Milone et al. (2012) that used the same RDb database) are not directly comparable to the present study because the studies vary in number and type of events analyzed, duration of records, type and height of pastures, recording procedures and devices, and validation methods. Furthermore, the data employed in those studies are not available for numerical experimentation. On the other hand, the remarkable capacity for event detection by CBRTA implies that further classification of ingestive events may not compromise the ability of an algorithm to efficiently detect ingestive events. Regarding the event classification stage, to assess the robustness of CBRTA, we decided to evaluate the performance of CBRTA for two sets of parameters, applied on the first database (MDb). The recognition rate averaged 84% when CBRTA was used with the best set of parameters for a partition (not further used for testing purposes) within the same database, and a performance rate of 79% when it was used with the best set of parameters for a partition (not further used for testing purposes) of a different database. This result shows that the algorithm seems to be robust to databases with large differences. For the second database (RDb), the proposed algorithm achieved a recognition rate of 77% on average, while the CBHMM method averaged 79.5% over all pastures. The best results achieved by CBHMM method is due to the use of a more complex modeling technique (hidden Markov models), which allows to capture more accurately the dynamics of the sounds and extract more information. However, this small performance improvement is achieved at expenses of a higher computational cost, as it can be appreciated in Appendixes A and B. Overall, the best results were seen for tall pastures reaching 79% and 78% for alfalfa and fescue respectively, while for short fescue a 77% was obtained. An additional deterioration of 5% in the recognition rate was detected for short alfalfa, which is consistent with previous findings by Milone et al. (2012). A plausible explanation for this is that short alfalfa plants have higher proportion of stems over leaves than tall alfalfa and fescue, and cows can produce bite sounds with lower amplitude, increasing confusion between events. Consequently, sound recordings for short forages, particularly alfalfa, may have a lower signal-to-noise ratio that can introduce errors in the classification of events. In the same way as for the first database good results for the classification of chews were obtained, which is a good sign for identification of rumination activities. Moreover, some degree of confusion between bites and chew-bites was also observed. That could be ameliorated by incorporating new sound features like a measure of symmetry of the event or information about the sequence of events.

Finally, as shown in the flowchart of the algorithm (Figure 3), any detected event that is not classified as chew-bite, bite or chew, is treated as noise event. However, potential insertions (false positive event) can occur when a given noise event is indeed misclassified as false chew, bite or chew-bite. To assess the odds for misclassified insertions, all likely insertions (false positive events due to noise) performed by CBRTA were further examined with the HTK performance tool. This analysis showed a low number of insertions, which has two plausible explanations. First, acoustic monitoring was conducted with directional microphones (sensing only in one direction) facing inward onto the forehead of cows and covered by a rubber foam (Milone et al., 2009). Furthermore, this instrumentation was made to minimize environmental noise (i.e. wind) to avoid the use of stronger high-pass filters, that otherwise will remove important information of sound signals. Second, the use of a low-pass filter with cutoff frequency of 5.5 Hz (or 3.5 Hz depending of database) for computing envelope was preferred over other filtering options. Noise is generally characterized as a non-stationary signal with high energy at high frequencies, and it would be expected that any noise energy that matches the frequency band of interest will have minimal influence or interference on both, detection and classification tasks.

6. Conclusions

It has been demonstrated the importance of acoustic monitoring for both detection and classification of ingestive events in grazing ruminants. Although this technique is very appropriate, it presents difficulties to automatically analyze large volume of high-quality audio signals by means of fast methods. These difficulties are usually related to computation load, power supply, data transfer and storage capacity. In this regard, the proposal was to develop an alternative algorithm that can get high accuracy for detection and classification, but with minimal computational cost.

The novel CBRTA algorithm was capable to combine very low computational cost with high accuracy for detection (up to 97.4%) and classification (up to 84.0%) of chews, bites and chew-bites in grazing dairy cattle. Furthermore, the linear computational complexity of CBRTA combined with the use of low sampling frequency and quantization level further minimized computational costs, which is a remarkable achievement in acoustics because it can lend to the application of very fast real-time execution for timely and accurate monitoring devices of grazing behavior. To the best of our knowledge, there are no other acoustic platforms that can be used for real-time analysis of sound signals in low cost embedded systems mounted on individual animals. The testing of CBRTA shows that with a sampling frequency of 4 kHz, good overall performance rate can be obtained at low computational cost. This suggests that the main energy for classification of ingestive events would be below to 2 kHz in a target signal, consistent with previous results obtained by Milone et al. (2012).

Given the demonstrated applicability of acoustic signals to assess herbage intake (Laca et al., 2000; Galli et al., 2011), future research steps must be focused on the automation of herbage intake measurements, as well as, on the application of acoustic monitoring as novel precision grazing management tool. Future equipment development must also focus on both integral applications that allow temporary storage or easy transfer of processed results via wireless network, and on intelligent power supply systems, that can assure long-time operation of acoustic devices and embedded microprocessors in field applications. The CBRTA algorithm has promising capability to facilitate these requirements.

Acknowledgements

The authors would like to thank Marcelo Larripa, Alejandra Planisich and Martín Quinteros from Facultad de Ciencias Agrarias, Universidad Nacional de Rosario, and staff of the Michigan State University's W.K. Kellogg Biological Station Dairy Farm for their assistance in animal management and gathering of acoustic data. This work has been funded by Agencia Nacional de Promoción Científica y Tecnológica, project PICT 2011-2440 "Desarrollo de una plataforma tecnológica para ganadería de precisión", Universidad Nacional del Litoral, PACT CAID 2011 “Señales, Sistemas e Inteligencia Computacional”, CAID 2011-525 and Universidad Nacional de Rosario, project 2013-AGR216. Additional support to Santiago Utsumi was provided by the USDA-National Institute of Food and Agriculture, project [MICL02224].

Appendix A. CBRTA complexity analysis

This appendix evaluates the computational cost of each step of the CBRTA algorithm, which depends on the number of samples n to be processed per second ($f_{CBRTA}(n)$). For this algorithm, the number of samples n to be processed per second depends on the sampling frequency. Hence, the number of operations required by each stage of the algorithm will depend on the task being performed:

1. **Signal rectification**: A simple pre-processing task that guarantee a positive sign for all samples. This task requires only a comparison and a multiplication ($2n$ operations/s).
2. **Signal filtering**: A second-order low pass filter is applied to the resulting signal to obtain the sound envelope. This filter can be implemented in two different ways: i) A second order infinite impulse response (IIR) filter that involves five multiplications and four additions ($9n$ operations/s) or ii) a finite impulse response (FIR) filter that involves P multiplications and P additions ($2Pn$ operations/s), where P is the number of taps employed by the filter. The use of one particular way of implementing the filter will depend on the main constraint of the implementation, such as computational efficiency for the FIR or numerical stability for the IIR.

3. **Signal subsampling**: To reduce the computational requirements (load and time) in the subsequent tasks, without losing accuracy, the sound envelope is subsampled from its original sampling frequency of 100 Hz. This task requires an addition and a comparison ($2n$ operations/s).
4. **Samples buffering**: The data stream generated in previous tasks is divided into short segments. From a computational point of view this task only involves counting of samples, which requires an addition (100 operations/s).
5. **Threshold generation**: The time-varying threshold $T(k)$ is computed through two steps: the computation of the peak expectation threshold (T_P), which requires five additions and one multiplication (600 operations/s), and the computation of the threshold $T(k)$, which requires one addition and two comparisons (300 operations/s). Therefore, the overall computational complexity of this task is 900 operations/s.
6. **Event detection**: This task only involves the comparison of the threshold $T(k)$ with the sound envelope, which implies a computational complexity of 100 operations/s.
7. **Properties computation**: This task computes the properties of the sound envelope for classification of events. The shape of a given event is quantified through computation of the number of changes in the sign of the envelope slope when its magnitude is bigger than the background noise. It requires one comparison and one subtraction (200 operations/s). The duration of the event is computed from the sound envelope by counting the number of samples when the envelope is bigger than the background noise. It requires one comparison and one

addition (200 operations/s). Finally, the maximum amplitude of the event is computed directly from the absolute value of sound over a window of time whose length is half of the duration of a typical chew-bite event. It only requires one comparison (100 operations/s).

8. **Event classification**: Using a set of five rules, based on the properties computed in the previous stage, the events are classified into chew, bite, chew-bite, silence and noise. The evaluation of a rule to classify a silence only requires the comparison of the sample counter k_T, which involve 100 operations/s. To evaluate the remaining rules, the algorithm checks the conditions that define each type of event. Therefore, the overall computational complexity for each of these rules is 500 operations/s. Since all rules are evaluated at every event, the overall complexity for this task is 2100 operations/s.

A linear complexity for CBRTA is given by the total number of operations per second that are required to be executed for an IIR low-pass filter implementation, as follows:

$$f_{CBRTA}(n) = 13n + 3700. \tag{A.1}$$

Appendix B. CBHMM complexity analysis

The cost of each step of the CBHMM algorithm presented by Milone et al. (2012) is evaluated as a function of the number of samples n to be analyzed per second ($f_{CBHMM}(n)$), where n depends on the sampling frequency. The corresponding system was implemented by the authors using the HTK toolkit (Young et al., 1997).

The signal will be analyzed using overlapped windows. Window duration w_L and window step w_S were defined as 60 ms and 40 ms, respectively. Regardless of the sampling frequency of input audio the number of windows n_w to be processed per second is

$$n_w = \left\lfloor \frac{s_L - w_L}{w_S} \right\rfloor + 1 = \left\lfloor \frac{1000\ \ ms - 60\ \ ms}{40\ \ ms} \right\rfloor + 1 = 24\ \ windows \qquad \text{(B.1)}$$

where s_L is the duration of the segment of signal to analyze. The number of samples n_S to be processed per window depends on the number of samples to be analyzed as

$$n_S = \frac{w_L}{s_L}\, n. \qquad \text{(B.2)}$$

Recognition processes can be separated into two main stages: i) feature extraction and ii) classification. During the feature extraction stage, each window is analyzed with the same exact processes. The following complexity analysis will be done for a single window of n_S samples:

1. **Pre-emphasis filter**: a simple pre-processing operation emphasizes the signal by applying a first order difference equation, that involves an addition and a multiplication ($2n_S$ operations).
2. **Windowing**: a Hamming window function is applied to pre-processed signal. This operation requires a multiplication for each sample of the window (n_S operations).
3. **Window energy**: is a numeric value obtained from windowed signal that will be part of the feature vector. It requires $2n_S$ operations.

4. **Fourier transform**: the windowed signal is transformed using a fast Fourier transform, and magnitude is then taken. Complexity of these computations are $n_S\ log(n_S)$ and n_S operations, respectively.
5. **Filterbank analysis**: is a simple transform based on a bank of triangular filters designed to give approximately equal resolution on a mel-scale. Each Fourier magnitude coefficient is multiplied by the corresponding filter gain and the results are then accumulated. Thus, each bin holds a weighted sum representing the spectral magnitude in that filterbank channel. Ten filters that spread between 0 and 500 Hz were selected by Milone et al. (2012). The complexity is a function of the maximum length of the filter F_{ML}, it is at most $20F_{ML}$ operations. Because it is clear that $F_{ML} \ll n_S$, then this operation should not be the most computationally expensive. It could be established a computational complexity of $20n_S$ operations as upper bound.
6. **Logarithm**: is applied to each channel parameter of the filterbank. This requires 10 operations (10 operations).
7. **Deltas**: a feature vector is composed by 22 elements is arranged by 10 log-filterbank parameters, window energy, deltas of log-filterbank parameters, and delta of energy. Deltas computation requires 11 additional operations.

The total number of operations required to extract features $f_{fe}(n_S)$ from a single window is

$$f_{fe}(n_S) = 21 + 26\ n_S + n_S\ log(n_S). \quad \text{(B.3)}$$

This number must be multiplied by n_w to obtain the complete number of operations in the feature extraction stage for one second of audio. The cost of classification stage is revised below under the same assumption that one second of audio must be processed.

Given the small number of models in this application (only 3: chew, bite and chewbite, without taking into account the silence model for simplicity) it is reasonable to suppose a similar complexity than in an isolated word recognition task. Also, one second of audio could contain only one event, due to the typical duration of masticatory events. Thus, to do isolated word recognition, the following steps must be performed: (i) generate a sequence of feature vectors corresponding to the audio, (ii) calculate the model likelihoods for all possible models, and, (iii) select the word whose model likelihood is highest.

Step (i) was already addressed in feature extraction stage. To perform step (ii) the Viterbi algorithm is used. This algorithm requires on the order of VQ^2T computations, where V is the number of words, Q is the number of states in each model, and T is the length of the feature vectors sequence (Rabiner and Juang, 1993). Since $V = 3$ (chew, bite and chewbite), $Q = 4$ and $T = 24.5$ (number of windows per second), the viterbi computations needed are $VQ^2T = 1,176$. Each Viterbi computation requires one multiplication, one addition, and a likelihood calculation (at least $M(d + d^2)$ operations (Duda, Hart, pp. 111), where $d = 22$ is the number of features, and the number of mixed gaussians is $M = 90$). Then, the operations needed in step (ii) are $VQ^2T(2 + M(d + d^2)) = 53,557,392$. Operations performed in step (iii) are just 3 comparisons to obtain the highest likelihood. Therefore, the number of operations performed in classification stage is determined by step (ii).

The algorithm CBHMM shows a superlinear complexity where the total number of operations per second of signal required to execute this algorithm is the sum of feature extraction and classification stage costs, as follows:

$$f_{CBHMM}(n) = 24\left\{21 + 26\,\frac{w_L}{s_L}\,n + \frac{w_L}{s_L}\,n\,log\left(\frac{w_L}{s_L}\,n\right)\right\} + 53,557,392. \tag{B.4}$$

References

Alkon, P. U., Cohen, A., 1986. Acoustical biotelemetry for wildlife research: a preliminary test and prospects. Wildlife Society Bulletin 14 (2), 193–196.

Alkon, P. U., Cohen, Y., Jordan, P. A., 1989. Towards an acoustic biotelemetry system for animal behavior studies. The Journal of Wildlife Management 53 (3), 658–662.

Christov, I. I., 2004. Real time electrocardiogram QRS detection using combined adaptive threshold. BioMedical Engineering OnLine 3 (1), 28–37.

Clapham, W. M., Fedders, J. M., Beeman, K., Neel, J. P., 2011. Acoustic monitoring system to quantify ingestive behavior of free-grazing cattle. Computers and Electronics in Agriculture 76 (1), 96–104.

De Boever, J., Andries, J., De Brabander, D., Cottyn, B., Buysse, F., 1990. Chewing activity of ruminants as a measure of physical structure—a review of factors affecting it. Animal Feed Science and Technology 27 (4), 281–291.

Delagarde, R., Caudal, J.-P., Peyraud, J.-L., 1999. Development of an automatic bitemeter for grazing cattle. Annales de Zootechnie 48 (5), 329–339.

Duda, R. O., Hart, P. E., Stork, D. G., 2001. Pattern classification, second ed. John Wiley and Sons.

Duizer, L., 2001. A review of acoustic research for studying the sensory perception of crisp, crunchy and crackly textures. Trends in Food Science & Technology 12 (1), 17–24.

Galli, J. R., Cangiano, C. A., Demment, M., Laca, E. A., 2006. Acoustic monitoring of chewing and intake of fresh and dry forages in steers. Animal Feed Science and Technology 128 (1), 14–30.

Galli, J. R., Cangiano, C. A., Milone, D. H., Laca, E. A., 2011. Acoustic monitoring of short-term ingestive behavior and intake in grazing sheep. Livestock Science 140 (1), 32–41.

Hodgson, J., Illius, A. W., et al., 1996. The Ecology and Management of Grazing Systems. CAB international, England.

Laca, E., Ungar, E., Seligman, N., Ramey, M., Demment, M., 1992. An integrated methodology for studying short-term grazing behaviour of cattle. Grass and Forage Science 47 (1), 81–90.

Laca, E., WallisDeVries, M., et al., 2000. Acoustic measurement of intake and grazing behaviour of cattle. Grass and Forage Science 55 (2), 97–104.

Milone, D., Rufiner, H., Galli, J., Laca, E., Cangiano, C., 2009. Computational method for segmentation and classification of ingestive sounds in sheep. Computers and Electronics in Agriculture 65 (2), 228–237.

Milone, D. H., Galli, J. R., Cangiano, C. A., Rufiner, H. L., Laca, E. A., 2012. Automatic recognition of ingestive sounds of cattle based on hidden Markov models. Computers and Electronics in Agriculture 87 (1), 51–55.

Navon, S., Mizrach, A., Hetzroni, A., Ungar, E. D., 2013. Automatic recognition of jaw movements in free-ranging cattle, goats and sheep, using acoustic monitoring. Biosystems Engineering 114 (4), 474–483.

Rabiner, L., Juang, B.-H., 1993. Fundamentals of speech recognition. Prentice-Hall.

Sauvant, D., 2000. Granulométrie des rations et nutrition du ruminant. Production Animales-Paris-Institut National de la Recherche Agronomique 13 (2), 99–108.

Tani, Y., Yokota, Y., Yayota, M., Ohtani, S., 2013. Automatic recognition and classification of cattle chewing activity by an acoustic monitoring method with a single-axis acceleration sensor. Computers and Electronics in Agriculture 92 (1), 54–65.

Ungar, E. D., Rutter, S. M., 2006. Classifying cattle jaw movements: comparing IGER behaviour recorder and acoustic techniques. Applied Animal Behaviour Science 98 (1), 11–27.

Watt, L., Clark, C., Krebs, G., Petzel, C., Nielsen, S., Utsumi, S., 2015. Differential rumination, intake, and enteric methane production of dairy cows in a pasture-based automatic milking system. Journal of Dairy Science 98 (10), 7248–7263.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., et al., 1997. The HTK book. Vol. 2. Entropic Cambridge Research Laboratory Cambridge.