Eur. Phys. J. Spec. Top.
https://doi.org/10.1140/epjs/s11734-021-00405-5

THE EUROPEAN PHYSICAL JOURNAL SPECIAL TOPICS

Check for updates

Regular Article

# Neural networks that locate and identify birds through their songs

Roberto A. Bistel[1,2], Alejandro Martinez[2], and Gabriel B. Mindlin[1,2,a]

[1] Departamento de Física, FCEyN, Universidad de Buenos Aires, Buenos Aires, Argentina
[2] IFIBA, CONICET, Buenos Aires, Argentina

Received 20 July 2021 / Accepted 16 December 2021


1 **Abstract** In this work, we present a set of algorithms that allow the location and identification of birds through their songs. To achieve the first objective, neural networks capable of reconstructing the position
2 of the subject are trained from a set of differences in the arrival times of a sound signal to the different microphones in an array. For the second objective, a dynamical system is used to generate surrogate songs, similar to those of a given set of subjects, to train a neural network so that it can classify subjects. Taken together, they constitute an interesting tool for the automatic monitoring of small bird populations.

## 1 Introduction

In recent years, machine learning and deep learning techniques have made it possible to attack a multiplicity of problems that until recently were prohibitively complex. In ecology, for example, one area of interest is the monitoring of animal populations. Studies in these areas can be facilitated, in the case of vocally active animals, by the automatic processing of the sounds that the animals make. Particularly in the case of birds, in the past few years much progress has been made in the automatic recognition of species through song, which has meant an important advance in the monitoring of avian biodiversity [1–4]. The convergence of two factors has been key to solving this problem. The first was the development of our calculation capacity, which has allowed the application of techniques such as deep learning neural networks to carry out classification tasks. The second factor has been the creation of international sound repositories such as Xeno-canto, from which it was possible to extract the enormous number of samples necessary to train the networks that perform the classification [5,6].

A problem somehow linked to the previous one is the localization and identification of individual wild birds through their vocalizations. This is relevant if you are looking to monitor the social behavior of a small population, which may be relevant for example, in the case of threatened species. This type of monitoring is also of interest in the framework of studying ethological processes such as the acquisition of song. In oscine birds, the song plays a fundamental role in a variety of social interactions, from territorial defense to partner selection. Wild birds under laboratory conditions show a limited behavioral response. That is why it is ideal to study these birds in their natural habitat, in which they show their complete behavioral repertoire. Much of this natural behavior takes place in a visually challenging landscape, such as, open foliage-free spaces. For this reason acoustic localization play an important complement in the ethological study of birdsong, providing a spatial context to the social interactions involving vocalizations.

The identification of subjects through song presents important challenges, particularly if one aspires to use methods such as neural networks, which were successful in identifying species. One of these challenges is the size of the samples that can be aspired to obtain, such as to train a network to identify a subject. Typically, it is possible to achieve the continuous registration of a set of songs and conclude that they come from a subject. But unless the individuals are ringed, and the visual code of the vocalizing subject can be visualized and identified, it is not possible to put together separate records and assign them to a single individual. For this reason, it is difficult to train a neural network with songs from a subject: the bases of songs attributable to a subject in the field are usually formed by a few examples [7,8].

In this work, we present a set of algorithms capable of locating birds through their vocalizations and identifying the vocalizing subject through certain specific patterns of their song. The locator algorithm begins with the processing of the acoustic signals, corresponding to recording a song by means of an array of four microphones. Taking the difference in arrival times at these different microphones, a neural network previously trained with artificially generated time differ-

[a] e-mail: gabo@df.uba.ar (corresponding author)



SPI Journal: 11734 MS: 0405 ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:2021/12/22 Pages: 10 Layout: EPJST

ences reconstructs the position of the sound source. On the other hand, the acoustic pattern identifier algorithm consists of a neural network capable of taking the image of a spectrogram corresponding to a song and classifying it among a set of pre-established classes. To train a neural network so that it can identify the acoustic patterns of a subject, it is trained with the images of the spectrograms corresponding to synthetic songs that emulate the real birdsong of a group of subjects [9,10]. In this way, it is possible to generate, from a few songs per subject, many surrogate songs capable of training the classification network.

This method of classifying acoustic patterns is used to classify individuals in those species in which it is required the exposure to a tutor to learn to vocalize, managing to crystallize one or more songs of their own, typically consisting of some combination of whistles characteristic of the species. An example is explored in this work, *Zonotrichia capensis*. This is a South American bird that needs an exposure to a tutor to sing. After a period of learning, it ends up incorporating a song. In exceptional cases, it can incorporate two or even three different themes [11–13]. To illustrate how these algorithms operate, in this work we train a neural network using surrogate synthetic songs to distinguish between a set of six different examples of *Zonotrichia capensis* songs. Applying the localization method, we find that three of the analyzed patterns actually corresponded to three songs generated by a single individual. Subsequent filming allowed to validate the result, highly unexpected since, according to the literature, only one out of approximately 500 specimens of this species can generate three different songs [11,12].

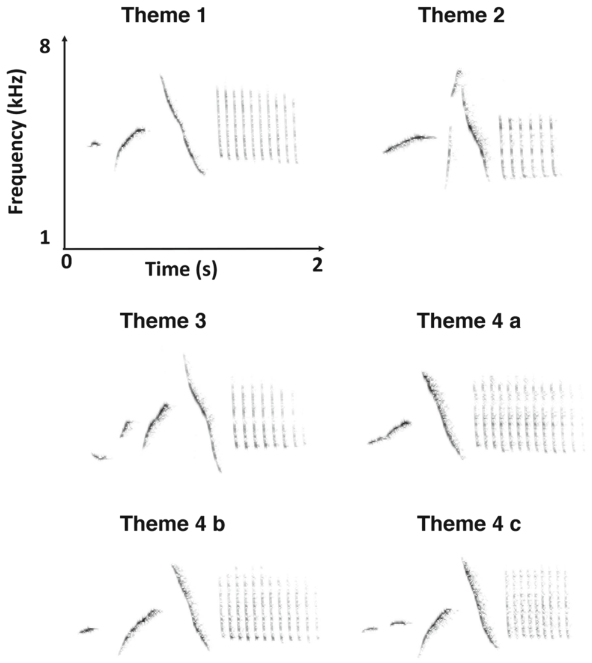



**Fig. 1** Six themes analyzed in this work, taken in four different places. The recordings were made with a sampling frequency of 44.1 kHz. Each spectrogram was found using a Gaussian window (standard deviation of 128 points), processing segments of 1024 samples, with successive overlaps of 512 samples. For the visualization of the spectrograms, a clipping of less than 1/600 of the maximum value of the spectrogram has been considered

## 2 Identification of themes using neural networks

The rufous-collared sparrow, or chingolo (*Zonotrichia capensis*) is a highly territorial songbird, which acquires its song after being exposed, as a juvenile, to a tutor. His song is a sequence of syllables that he sings for a period of between 2 and 3 s and is made up of two parts. The first is an introductory sequence of between 1 and 5 syllables whose frequency is modulated. This first part is known as a theme, and each individual typically has a characteristic one, although there are individuals capable of singing two or three different themes. The second part is made up of a trill; a rapid repetition of identical syllables [11–14]. Figure 1 shows a set of spectrograms representative of the song produced by the chingolos in this study. We analyzed 52 songs corresponding to six different themes, recorded in four different sites of *Parque Pereyra Iraola (Buenos Aires Province, Argentina)*.

When we need to automatically identify species by song, there are databases with hundreds of examples of song by species that can be used to train a neural network to perform the task. On the contrary, if the challenge is to identify individuals, for each non-ringed subject, it is only possible to assume as songs of the individual those recorded in a continuous recording. Thus, it is difficult in principle to obtain more than a few dozen examples putatively corresponding to a given individual. For this reason, it is an important challenge to train a network to identify subjects. Neural networks are extraordinary algorithms capable of classifying patterns (for example, the image of a spectrogram corresponding to a song), but the enormous number of parameters to be adjusted (the connections between neurons, precisely), requires a significant number of previously classified patterns to train the network [15].

To overcome this difficulty, in a previous work, it was proposed the training of the classifying neural network by means of a set of synthetic songs. They were generated by integrating a physical model of avian song production, which summarizes the biophysics of the avian vocal organ [8]. These solutions have been shown to be good enough mimics to achieve responses in highly selective neurons to the bird's own song, when used as auditory stimuli [9,16]. Using the few songs obtained for each individual and estimating the variability of the initial and final values of the frequencies of the syllables of each song, we generated synthetic songs to train a neural network.

### 2.1 Description of the model for synthesizing song

The model that we will use to generate the synthetic songs used to train our network describes the way in which song is generated in birds. Song is generated at the syrinx, which is a structure that supports two pairs of lips, at the junction between the bronchi and the trachea. These pairs of lips go into an oscillatory mode when a sufficiently strong flow of air passes between them, just like human vocal cords when a voiced sound is emitted. The oscillations produced modulate the air flow and generate the sound that is emitted [17].

The basic physiological parameters that the birds need to control to generate the song are the pressure of the air sac, which controls the intensity of the air flow through the lips, and the physiological instructions sent to the syringeal muscles. The configuration of the syrinx, which has a certain elasticity, affects the stretching of the lips and, therefore, the fundamental frequency of the labial oscillations [17].

The lips are assumed to be in a stationary position when the bird is silent. Once the parameter representing air sac pressure is increased, a threshold for oscillatory motion is reached. If the problem parameters remain in the phonation region of the parameter space, the airflow is modulated, and sound is produced. As the pressure decreases, the sound eventually stops (that is, the syllable ends). A qualitative change in dynamics when the parameters are varied is known as a bifurcation. Near the values of the parameters where the bifurcation occurs, the model can be transformed into simple equations that describe the dynamics of the system. For the chingolo, the system of equations that describes the dynamics of the lips is the one shown in Eq. (1) [18].

$$\begin{cases} \frac{dx}{dt} = y \\ \frac{dy}{dt} = k\gamma^2 x - \gamma x^2 y + \beta\gamma y \end{cases} \tag{1}$$

In Eq. (1), $x$ represents the midpoint position of the lips; $k$, $\beta$ are parameters of the system; while $\gamma$ represents the time scale of the system. The generation of sound with this dynamic of the lips, occurs when the pressure at the entrance of the trachea $p_i$, is shown in Eq. (2).

$$\begin{cases} p_i(t) = A\frac{dx(t)}{dt} + p_{back}(t - \frac{L}{c}) \\ p_{back}(t) = -r p_i(t - \frac{L}{c}) \end{cases} \tag{2}$$

In Eq. (2), $A$ is the average area of the lumen; $L$ is the length of the trachea; $c$ is the speed of sound in the medium; while $r$, is the reflection coefficient at tracheal exit. This leads to the pressure at the exit of the trachea $p_o = (1-r)p_i(t - \frac{L}{c})$, which forces a Helmholtz oscillator representing the oropharyngeal–esophageal cavity (OEC).

The OEC behaves like a signal filter, and its operation is modeled through the set of equations (3) [19].

$$\begin{cases} \frac{di_1}{dt} = i_2 \\ \frac{di_2}{dt} = -\frac{i_1}{cL_1} - \left(\frac{r_d}{L_2} + \frac{r_d}{L_1}\right) i_2 + \left(\frac{1}{cL_1} + \frac{r_2 r_d}{L_1 L_2}\right) i_3 + \frac{\frac{dp_0}{dt}}{L_1} + (\frac{r_2 r_d}{L_1 L_2})p_o \\ \frac{di_3}{dt} = -\left(\frac{L_1}{L_2}\right) i_2 - \left(\frac{r_d}{L_2}\right) i_3 + \left(\frac{1}{L_2}\right) p_0 \end{cases} \tag{3}$$

The set of equations (3) has been rewritten in such a way that the dynamics of the Helmholtz oscillator with aperture is represented through an equivalent circuit. These equations are derived in [19], the final sound being proportional to the value of the variable $i_3$. The parameters used for the generation of synthetic song are $(L_1, L_2, r_2, r_d, c) = (1/20, 1/10^4, 0.5 \times 10^7, 24 \times 10^3, 5/350{\times}10^8)$.

For many species, the various acoustic modulations in song are translated into a set of basic physiological instructions called "gestures" [20]. In the case of the *Zonotrichia capensis*, these acoustic modulations can be defined using three frequency modulation patterns: sinusoidal, linear, and exponential down sweep. The parameters for each modulation pattern are presented in Table 1.

To synthesize the song using the model, the modulation pattern of each syllable is identified, and the necessary parameters (Table 1) for its reproduction are found. Then for each syllable a list of fundamental frequencies is generated. The values of the system parameter $k$, which allow the generation of songs with the fundamental frequencies $w$ satisfy: $k = 6.5 \times 10^{-8} w^2 + 4.2 \times 10^{-5} w + 2.6 \times 10^{-2}$. The relationship between $k$ and $w$ was obtained through a series of numerical simulations in the parameter space of the model, varying the values of $k$, and computing for each simulation the fundamental frequency of the synthesized song $w$. Then, we proposed a polynomial relationship between $w$ and $k$, and used the list of pairs $(k, w)$ to compute the coefficients of the polynomial through a regression [18]. Thus, the list of fundamental frequencies is transformed into the parameters that the model uses to synthesize a realistic copy of the song. Using the synthetic song generation model, the spectral content of the sound source

**Table 1** Basic patterns for the gestures used to synthesize the song of the *Zonotrichia capensis*

| Modulation pattern | Frequency | Parameters |
|---|---|---|
| Sinusoidal | $w(t) = w_f + (w_i - w_f)(\frac{t-t_i}{t_f-t_i})$ | $w_i, w_f, t_i, t_f$ |
| Linear | $w(t) = w_{av} + Asin(\alpha_i + (\alpha_f - \alpha_i)(\frac{t-t_i}{t_f-t_i})$ | $w_{av}, A, \alpha_i, \alpha_f, t_i, t_f$ |
| Exponential | $w(t) = w_f + (w_i - w_f)e^{-\frac{3(t-t_i)}{(t_f-t_i)}}$ | $w_i, w_f, t_i, t_f$ |

is automatically reproduced, correctly filtered by the trachea and the OEC. In other words, we fit the fundamental frequencies, and the spectral content is automatically reproduced by the model. This is particularly important when the method is applied to species with harmonically rich sounds.

We proceeded to integrate the model a large number of times, varying the values of the parameters presented in Table 1 to reproduce the basic gestures [18]. The parameters characterizing the song (the initial and final values of the fundamental frequency for each syllable, the duration of each syllable, and the timing between syllables) varied very little across different repetitions of the song; never more than 3%. The variations of the values in the parameters were obtained from a Gaussian distribution with the means and standard deviations calculated from the song examples for each of the six themes of interest. We used ten songs to estimate the parameters for all the themes but *Theme 4 c*, for which we had only two songs.

Thus, a large number of surrogate spectrograms are generated, all of them differing in random parameters that are consistent with the biological variability that exists between different songs produced by a single individual [8]. These surrogate spectrograms become the training set, the validation set and artificial testing set for the neural network for identifying individuals. We generated, for each of the six different themes, 3500 spectrograms as surrogate data. From this set of synthetic spectrograms images, 2000 were randomly taken for model training, 1200 for validation, and 100 for model testing. None of these sets included any images of the actual spectrograms of the chingolos corresponding to the field recordings. Figure 2 shows some of the spectrograms generated from the dynamic model, for each of the themes of interest. The neural network training procedure was performed with the same hyper parameters and network structure shown in [8].

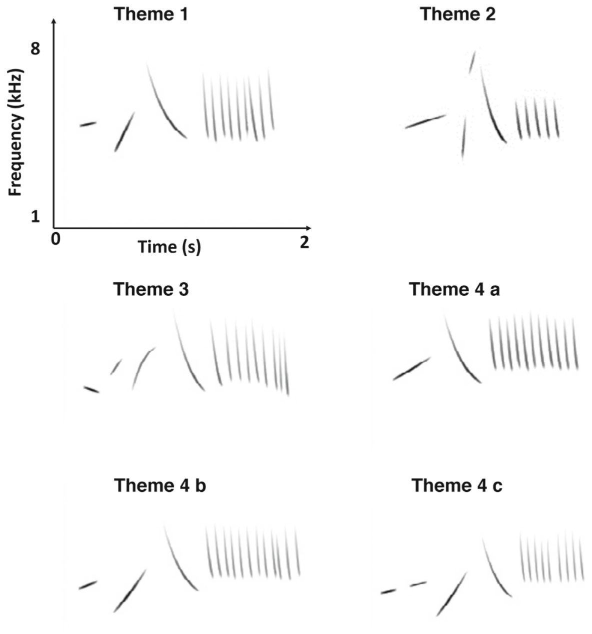



**Fig. 2** Some of the spectrograms generated from the dynamic model for each of the six themes of interest

## 2.2 Description of the neural network used to identify themes

The theme identification neural network takes the spectrograms of the songs as an image and classifies them with a given probability into one of the six themes of interest.This neural network is composed of four 2D convolutional layers that alternate with four MaxPooling layers. The network features a final pair of tightly connected layers. The 2D convolutional layers have sizes of 8, 16, 16 and 32 respectively, which are obtained from their respective inputs, after performing a convolution with $3 \times 3$ size windows. All MaxPooling layers perform a dimensionality reduction by a factor of 2, making the images smaller. This allows to reduce the computational cost, minimize the possibility of overfitting and increase the abstraction on the input data. The final two tightly connected layers consist of 1024 and 6 units, respectively. This last layer has 6 units since it is the number of classes to identify in our problem.

In the network, another tool to avoid overfitting is to establish restrictions on the connection values (weights) of the neurons, so that they take small values. The procedure, known as regularization, is implemented by adding a cost to the network loss function, whenever the weights take large values. In our network, the regularization parameter was established as $l2 = 0.001$. In addition, with the same objective of avoiding overfitting, they were made to drop some weights at random (setting their values to zero). The dropout value was set to 0.5, and the learning rate was established at $10^{-4}$. The spectrograms, used as images to train the network, were grayscale, with a size of $300 \times 200$ pixels. The batch size used was 10 units, while the training was carried


SPI Journal: 11734 MS: 0405 ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:2021/12/22 Pages: 10 Layout: EPJST

**Table 2** The confusion matrix for the classification of synthetic spectrograms images

| | *Theme 1* | *Theme 2* | *Theme 3* | *Theme 4 a* | *Theme 4 b* | *Theme 4 c* |
|---|---|---|---|---|---|---|
| *Theme 1* | 100 | 0 | 0 | 0 | 0 | 0 |
| *Theme 2* | 0 | 100 | 0 | 0 | 0 | 0 |
| *Theme 3* | 0 | 0 | 100 | 0 | 0 | 0 |
| *Theme 4 a* | 0 | 0 | 0 | 100 | 0 | 0 |
| *Theme 4 b* | 0 | 0 | 0 | 0 | 100 | 0 |
| *Theme 4 c* | 0 | 0 | 0 | 0 | 0 | 100 |

**Table 3** The confusion matrix for the classification of the spectrograms images of the real songs recorded

| | *Theme 1* | *Theme 2* | *Theme 3* | *Theme 4 a* | *Theme 4 b* | *Theme 4 c* |
|---|---|---|---|---|---|---|
| *Theme 1* | 7 | 1 | 0 | 0 | 2 | 0 |
| *Theme 2* | 0 | 10 | 0 | 0 | 0 | 0 |
| *Theme 3* | 1 | 0 | 9 | 0 | 0 | 0 |
| *Theme 4 a* | 0 | 2 | 0 | 8 | 0 | 0 |
| *Theme 4 b* | 4 | 2 | 0 | 0 | 4 | 0 |
| *Theme 4 c* | 0 | 1 | 0 | 0 | 0 | 1 |

**Table 4** *Precision*, *Recall* and *f1_score* for the classification of the real songs recorded

| | *P* | *R* | *f1_score* | *Support* |
|---|---|---|---|---|
| *Theme 1* | 0.58 | 0.70 | 0.64 | 10 |
| *Theme 2* | 0.62 | 1.00 | 0.77 | 10 |
| *Theme 3* | 1.00 | 0.90 | 0.95 | 10 |
| *Theme 4 a* | 1.00 | 0.80 | 0.89 | 10 |
| *Theme 4 b* | 0.67 | 0.40 | 0.50 | 10 |
| *Theme 4 c* | 1.00 | 0.50 | 0.67 | 2 |
| *Macro avg* | 0.81 | 0.72 | 0.73 | 52 |

out for 20 epochs, with 220 steps per epoch. For the validation, 80 steps were used per epoch. The network uses the Keras library, and in particular the ImageDataGenerator class. In this way, the images become tensors. Each image was normalized with a factor of 255.

## 2.3 Results in the identification of themes

The trained network was asked to classify 100 songs taken randomly, which were not used in previous steps of the training and validation model. To evaluate the performance of the neural network in the classification of these 100 synthetic spectrograms images, we calculated the confusion matrix. Table 2 presents the results obtained for the confusion matrix. In the confusion matrix, each row corresponds to a class (theme in our case), while the column represents the predicted class.

The performance of the neural network is obtained through the classification of spectrogram images corresponding to real songs. For this test, we used the 52 real songs recorded. Noise reduction filters and band pass filters between 1.5 and 8 kHz were applied to the field recordings. The spectrograms corresponding to each recording were calculated using the same parameters as those corresponding to the spectrograms of the synthetic songs. Each of these spectrograms was used as input to the trained network. Table 3 presents the confusion matrix obtained for the classification of the spectrograms images of the real songs recorded.

The network tends to incorrectly classify the songs from *Theme 4 b* with those from *Theme 1*. This is due to the similarity that exists between statistical parameters and the patterns of frequency modulation in this two themes, as shown in Fig. 1. The main difference between these two themes is the duration and frequency value of the first syllable, varying very slightly between them. The neural network is not able to differentiate this characteristic in some of the real spectrograms. In Table 3 it is also shown that one of the two real songs corresponding to *Theme 4 c*, is incorrectly classified as belonging to *Theme 2*.

From the confusion matrix it is possible to calculate a group of metrics that summarize the behavior of the network in the classification of each of the classes. Typical values that are calculated are *Precision*, *Recall*, and *f1_score*. *Precision* ($P$) indicates the ratio between correctly predicted instances for a given class, and the all predicted labels for that class. The *Recall* ($R$) value indicates for all instances that should have an $X$ label, how many of them were correctly labeled. In turn, *f1_score* measures the bal-



SPI Journal: **11734** MS: **0405** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2021/12/22** Pages: **10** Layout: **EPJST**

ance between the *Recall* and *Precision* indices. Table 4 shows the values of these metrics, which were calculated from the confusion matrix presented in Table 3.

The lowest $P$ is reached for the *Theme 1* class with $P = 0.58$. The lowest *Recall* value is for *Theme 4 b* with $R = 0.4$, since, out of a total of 10 songs, only four were correctly classified. The mean value of $f1\_score$ was $f_1 = 0.73$. This value is considered acceptable, since the network was trained without ever being exposed to the spectrograms of the real songs of the field recordings. The network training process was performed ten times using the set of artificial spectrogram images. In all experiments the corresponding confusion matrices were constructed. The average values and standard deviations in the classification of the 52 real songs recorded was (*Precision*, *Recall*, and $f1\_score$): $P = 0.80 \pm 0.04$, $R = 0.71 \pm 0.02$, $f1\_score = 0.71 \pm 0.02$.

## 3 Location of sound sources by the method of time delays

In the case of the common chingolo, each subject typically has a characteristic theme. A small number of subjects can sing two different themes, and an even smaller number are capable of singing three different themes [11–13]. An automatic subject identification procedure using the song themes as a classification parameter, will lead to the identification of two or three different individuals whenever two or three themes are detected. If each song can be accompanied by an observer who verifies the identity of the subject, the problem is solved, but an automatic method based on recordings encounters an important limitation. One way to solve the problem is to record the sounds with a set of microphones, which allow to triangulate the position of the recorded songs. In this way, themes that can be associated with subjects capable of singing various themes will emerge as emitted from the same position. For this reason, we propose to develop a mechanism (equipment and algorithms) capable of estimating the position from which a specific song comes.

The strategy used to develop the sound locator is to simultaneously measure the sound generated by a source, by means of an array of microphones connected to a recorder. The microphones are in the array at certain positions $\mathbf{x_i}$, such that when a source at position $\mathbf{p}$ emits a signal at time $t_0$, then the source can be located.

In practice, since the sources are birds, the signal emission time $t_0$ is unknown. Then the data that can be extracted from the microphones is the relative arrival times between pairs of receivers. Obtaining the position from this information is known as location by time difference of arrival (TDOA: Time Difference of Arrival). Equation (4) represents the arrival time of the signal at microphone $i$, where $c$ is the speed of sound.

$$t_i = \frac{|\mathbf{p} - \mathbf{x_i}|}{c} + t_0 \tag{4}$$

The equations for the temporal differences in signal arrival between microphones correspond to Eq. (5).

$$t_i - t_j = \frac{|\mathbf{p} - \mathbf{x_i}|}{c} - \frac{|\mathbf{p} - \mathbf{x_j}|}{c} \tag{5}$$

From four spatially separated microphones, we have the minimum information necessary to reconstruct the position of a sound source in three dimensions [21–25]. There are algorithms that analytically calculate the position of the source from the position of the microphones and the time differences. These methods have a poor response to the presence of errors in the calculation of the temporal differences for the estimation of the sound source.

These errors can occur for different reasons. In the first place, there are those associated with the sampling frequency of the system. As sound travels at approximately 350 m/s, errors are accentuated when the distance traveled by sound between two consecutive measurements is comparable to the distance between microphones. Therefore, small microphone arrays produce time differences that can be very small and on the order of the sampling frequency range. Other sources of errors are related to the measurement of the audio signal in noisy environments, as well as the variability of the signal intensity, which affects the signal-to-noise ratio (SNR) of the recording.

An alternative to the analytical methods of calculating the temporal differences is to overdetermine the problem and carry out a regression from a set of data generated by means of numerical simulations. Regression can be done using deep learning and machine learning techniques. The strategy consists of exposing the system, during a previous training phase, to data from which the result is known. Thus, the position of hundreds of possible sound sources is modeled, and the temporal differences are found. Then, using deep learning and a neural network, you learn to recognize the position of the sound source.

In our case, the input is a vector of dimension $\binom{N}{2}$, where $N$ is the number of microphones. The output is a three-dimensional vector, which corresponds to the $x$, $y$ and $z$ positions of the sound source. The training of the model is carried out with a data set $E$, where for each combination of temporal differences $E_i$ we have the position of the source that generates those temporal differences.

For our estimation of sound source's position, we chose to bound the maximum error to 1 m, for sound sources at a distance of up to 20 m. This would allow us to identify a tree for these highly territorial birds. Since the system has to be small in size and easy to install, it was decided in a first stage that it should only be made up of four microphones. The microphones will be located in the same plane, on the surface of the ground, and at the ends of a square circumscribed in a circumference. In our measurements, a commercial *Zoom H6* recorder was used, which has up to six audio inputs that are recorded simultaneously.



SPI Journal: **11734** MS: **0405** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2021/12/22** Pages: **10** Layout: **EPJST**

### 3.1 The neural network used to localization

The neural network for the location of individuals takes as input parameters for training: the maximum radius $r$ in which it is desired to locate, the speed of sound $c$, the sampling frequency $f_s$ used in the recordings, and the positions of the microphones. With these parameters a set of artificial positions are generated by numeric simulations up to the maximum location radius indicated. For each artificial position, we computed the arrival time to each microphone. We used these times to calculate the difference in the arrival time to each pair of microphones. We added a uniform random error $\zeta$ between $\pm\frac{1}{f_s}$ to each time difference, to account for the uncertainties due to the sampling used in the recordings. The arrival time to each microphone is calculated using Eq. (6) as:

$$t_i = \frac{\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}}{c} + \zeta. \qquad (6)$$

The set of artificial positions generated and the differences in arrival time are randomly divided into the training data set and the validation data set.This is the $k$-fold cross validation method. The amount of data that passes to each set is determined by the value of the parameter $k$. The data are randomly distributed in $k$ groups of approximately the same size, $k-1$ groups are used to train the model and one of the groups is used as validation. This process is repeated $k$ times using a different group as validation in each iteration. The process generates $k$ estimates of the error, the average of which is used as the final estimate [15].

The neural network for the location of individuals uses a sequential model, composed of five dense layers, where the first four have 64, 128, 128 and 64 units. The last layer, which is the output layer, has 3 units, which correspond to the geometric positions $(x; y; z)$ of the sound source to be located. The activation function of each layer is the ReLU. The model was compiled using the RMSprop algorithm as optimizer, and the loss function parameter used is the mean square error ($mse$).

The neural network used in this work was trained for a maximum search radius of 20 m, with a total of $1.6 \times 10^5$ sources equispaced 0.1 m in the training radius. The sampling frequency was 44.1 kHz and a sound speed of 350 m/s. Four microphones located at the ends of a square with a side of 7 m were taken as signal receivers. The value of $k$, which divides the data between the training and validation groups, was set at $k = 3$. The network was trained with a batch size of 1 unit, for 1200 epochs.

To test the trained model, we used a set of 14,400 artificial positions. This corresponds to sources equally spaced 0.3 m in a radius of 18 m. The mean error in the location is $0.32 \pm 0.23$ m, with a maximum error of 2.62 m. The median error is 0.268 m. The percentage of values with an error greater than the mean is 38.40%, while with an error greater than 1 m is 2.0%.

### 3.2 Processing of the audio signals

To determine the temporal differences in the arrival of the signal to each pair of microphones, it is necessary to precisely find the beginning of a sound in each file corresponding to the microphone. The possibility of finding the onset of a sound through a threshold is ruled out, since measurements are made in the field. Therefore, recordings are variably affected by ambient noise and the occurrence of various audio signals simultaneously. In addition, as a result of the degradation of the signal, the sound reaches each microphone with different amplitude, making it impossible to carry out an analysis by determining maximums. All of this makes it difficult to obtain a signal where there are no different points that can be considered as the beginning of a certain sound [26,27].

To minimize errors in the calculation of temporal differences, microphones with equal sensitivity were used, and the gain of each channel was calibrated on the *Zoom H6* recorder. In addition, a pre-processing of the signal was performed. This pre-processing consists of applying noise reduction filters, and band-pass signal filters, which reduce the bandwidth to the frequencies of interest of the sound in question. In this way, ambient noise is reduced and overlap in time and frequency is limited, due to the existence of multiple sounds.

Each signal segment of interest was normalized in amplitude, and then a 12th-order Butterworth FIR-type band-pass filter was applied, with cut-off frequencies between 1 and 8 kHz. This bandpass filter has been implemented using the *sosfiltfilt* function from the *scipy* signal library in Python. A noise reduction filter is then applied to it using spectral subtraction. This filter estimates the instantaneous signal energy and the noise floor for each frequency interval, being used to calculate a gain filter with which to perform spectral subtraction. The filter implementation uses the *pyroomacoustics* library available for Python. The parameters used for this filter are a window width of 512 samples, a noise reduction value of 3 dB, a loopback value of eight samples, and an overestimate value of the filter's gain $\beta$ of 6 dB. After filtering the signal is normalized again in amplitude.

Then, for each signal segment where the sound occurred, the correlation function is determined, so that the value found corresponds to the number of samples necessary for the signals to be aligned [28–30]. The correlation function finds the similarity between two signals for all possible delays $\tau$, as in show in Eq. (7).

$$corr(\tau) = \sum_{t=0}^{N-1} s_1(t)s_2(t+\tau) \qquad (7)$$

Equation (8) shows that the peak of the correlation function occurs at the value that maximizes the similarity between the two signals, which is, in turn, the number of samples necessary for both signals to be aligned. Since the number of samples is related to the sampling frequency $f_s$ of the system, we then have the time dif-

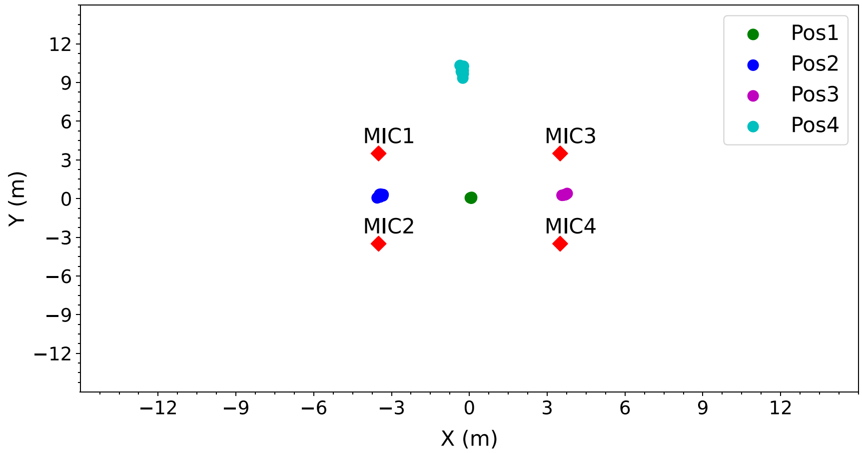



**Fig. 3** Test using metronomes in four locations

ference between each pair of microphones.

$$\tau_{est} = argmax(corr(\tau)) \quad (8)$$

To robustly determine temporal differences, it is necessary to accurately find the peak of the correlation function. To do this, the correlation must have a distinctive and prominent peak, corresponding to the signal of interest.

In signal processing, the onset of a sound is determined by calculating the statistical values of the signal. First, after filtering and normalizing the signal, the envelope of the signal is determined. This is done through the calculation of the absolute value of the Hilbert envelope. The envelope is smoothed with a Butterworth low pass filter with cutoff frequency 250 Hz and order 8. Then, the standard and mean deviation are calculated for a time window, traversing the signal in such a way that when the background noise is overcome, then there is an abrupt increase in the statistical parameters. This makes it possible to determine that the sound started at that moment and, therefore, the correlation between each pair of microphones can be calculated. The way to detect the distance from the background noise values is by finding the peak of the second derivative of the signal. The window width used for the calculation of the statistical parameters was 1024 samples. The calculation of the cross correlation was carried out using the correlate function of the *scipy* signal library in Python. A full correlation mode and a window width of 44,100 samples were used, which corresponds to 1 s of signal at a sampling frequency of 44.1 kHz.

### 3.3 Calibration using metronomes

The field tests for the calibration and experimental validation of the system were carried out using a metronome located for 10 seconds in pre-established positions. These positions correspond to the geometric center of the system (0; 0), (−3.5; 0), (3.5; 0) and (0; 10), where all positions are in meters. Figure 3 shows the results of calculating the positions from the audio recordings. For each position, a total of eight audio segments were analyzed, to which the differences in arrival time have been calculated.

The results in the location of the sound source are consistent with the application to be developed. Table 5 shows the statistical results of the calculation of the positions for the test corresponding to Fig. 3. The location error is less than 0.35 m, fulfilling the proposed objective of an error of less than 1 m. The standard deviation of the positions on each coordinate axis is less than 0.3 m, indicating a high repeatability of the algorithm. Therefore, the system developed for the location can be used to estimate the location of birds in the field.

## 4 Neural network for the localization of individuals

The system composed of the neural network for the identification of individuals and the neural network for the estimation of positions, was used to process a three field recordings (approximately 5 min of audio on each recording) from the site where it is known that there are chingolos that perform the *Theme 4 a*, *Theme 4 b* and *Theme 4 c*. The hypothesis tested is that some individual is capable of generating more than one theme pattern in his song. The four microphones used for recording were located at the ends of a square with a side equal to 14 m. The neural network for localization was trained with the same parameters of the network presented in Sect. 3.1.

The processing of these recordings made it possible to detect the presence of songs segments separated by 7–8 s, which corresponded to predictions of the neural network as corresponding to the *Theme 4 a*, *Theme 4 b* and *Theme 4 c*. Table 6 shows the prediction results returned by the identification network for a segment of three consecutive songs.

The network returns a series of values that can be interpreted as the probability that the predicted observation belongs to each of the possible classes. The highest probability represents the class predicted by the



SPI | Journal: 11734 MS: 0405 ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:2021/12/22 Pages: 10 Layout: EPJST

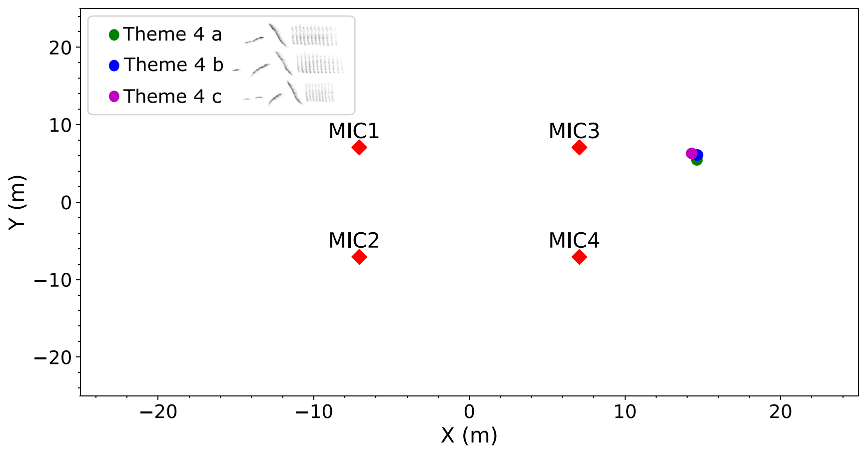
**Table 5** The results of the test described in Fig. 3

| | Median (m) | Error (m) | $\sigma$ (2D) (m) |
|---|---|---|---|
| Pos1 (0; 0) | (0.071; 0.058) | 0.091 | (0.014; 0.016) |
| Pos2 (−3.5; 0) | (−3.426; 0.264) | 0.274 | (0.067; 0.094) |
| Pos2 (3.5; 0) | (3.678; 0.3) | 0.348 | (0.054; 0.039) |
| Pos3 (0; 10) | (−0.258; 9.874) | 0.287 | (0.041; 0.295) |

**Table 6** Probability of each song of corresponding to a given theme

| | *Theme 1* | *Theme 2* | *Theme 3* | *Theme 4 a* | *Theme 4 b* | *Theme 4 c* |
|---|---|---|---|---|---|---|
| *Song 1* | 0.016 | 0.173 | 0.089 | 0.686 | 0.024 | 0.011 |
| *Song 2* | 0.356 | 0.192 | 0.027 | 0.010 | 0.401 | 0.013 |
| *Song 3* | 0.208 | 0.051 | 0.017 | 0.084 | 0.294 | 0.345 |

**Fig. 4** Estimated locations of the songs characterized as *Theme 4a*, *Theme 4b* and *Theme 4c*




network. As can be seen in Table 6, *Song 1* has a greater probability of belonging to *Theme 4 a* with a $P = 0.686$, while for *Song 2* it corresponds to *Theme 4 b* with $P = 0.401$, and for *Song 3* it is corresponds to *Theme 4 c* with $P = 0.345$. Given that *Song 2* and *Song 3* present probabilities of belonging to a class close to other classes, a visual inspection was carried out. The presence of these consecutive songs belonging to three different themes was verified counting syllables in the spectrograms of the field recordings. As there was little time separation between these songs, we proceeded to calculate the differences in the time of arrival at the microphones, to estimate the geographic location of the songs.

Figure 4 shows the location predicted by the network for *Songs 1*, *2* and *3* previously processed.

The estimated location of *Song 1* is (14.65 m; 6.08 m); for *Song 2* it is (14.28 m; 6.29 m); and for *Song 3* the position is (14.61 m; 5.46 m). Therefore, it can be said that the three patterns analyzed for *Theme 4 a*, *Theme 4 b* and *Theme 4 c*, actually correspond to three songs generated by a single individual. Subsequent video footage allowed the validation of the result, which is highly unexpected since, according to the literature, only one in 500 specimens of this species can generate three different themes [11].

## 5 Discussion

In the present work, we have described a set of algorithms capable of locating and identifying birds by their songs. The process of identifying songs themes was supported by the construction and training of a neural network. Unlike what happens with the identification of avian species through song, the identification of individual subjects required the generation of a large number of surrogate songs, which were generated by synthesizing an avian vocal production model. These models, based on the dynamic mechanisms associated with the generation of labial oscillations in the vocal apparatus, were able to generate songs that were realistic enough for the networks trained with them to be able to later identify true songs.

The process of identifying subjects through themes included the construction of an algorithm capable of reconstructing, from recordings, the position of the speaking subject. The algorithm uses a set of times as a way of calculating the relative times of arrival of a sound signal to different microphones connected to the same recording device.

As an example of our workflow, with the combined use of an automatic system for the identification of songs themes and a sound localization system, we were

able to find an individual capable of executing multiple
3 themes, a rare event in this species (see a video in [31]).
4 In any case, the algorithms presented here constitute a powerful tool for the automatic monitoring of avian populations through their vocalizations; a tool that can play an important role in the study and monitoring of small populations, particularly those corresponding to threatened species.

**Data availability statement** This manuscript has associated data in a data repository. [Authors' comment: ...]

## References


1. A. Thakur, P. Rajan, IEEE J. Sel. Top. Signal Process. **13**(2), 298–309 (2019)
2. D. Stowell, M.D. Wood, H. Pamuła, Y. Stylianou, H. Glotin, Methods Ecol. Evol. **10**(3), 368–380 (2019)
3. Z.J. Ruff, D.B. Lesmeister, C.L. Appel, C.M. Sullivan, Ecol. Indicators **124**, 107419 (2021)
4. Y. Maegawa, Y. Ushigome, M. Suzuki, K. Taguchi, K. Kobayashi, C. Haga, T. Matsui, Ecol. Inform. **61**, 101164 (2021)
5. S. Kahl, C.M. Wood, M. Eibl, H. Klinck, Ecol. Inform. **61**, 101236 (2021)
6. K. Nagy, T. Cinkler, C. Simon, R. Vida, in: *2020 IEEE SENSORS*, (2020), pp. 1–4
7. D. Stowell, T. Petrusková, M. Šálek, P. Linhart, J. Roy. Soc. Interface **16**, 153 (2019)
8. P.L. Tubaro, G.B. Mindlin, Chaos Solitons Fract. X **2**, 100012 (2019)
9. G.B. Mindlin, Chaos Interdiscip. J. Nonlinear Sci. **27**(9), 092101 (2017)
10. A. Amador, Y.S. Perl, G.B. Mindlin, D. Margoliash, Nature **495**(7439), 59–64 (2013)
11. F. Nottebohm, Condor **71**(3), 299–315 (1969)
12. F. Nottebohm, R.K. Selander, Condor **74**(2), 137–143 (1972)
13. C. Kopuchian, D.A. Lijtmaer, P.L. Tubaro, P. Handford, Anim. Behav. **68**(3), 551–559 (2004)
14. P.L. Tubaro, Ph.D. tesis, Universidad de Buenos Aires, Facultad de Ciencias Exactas y Naturales, (1990)
15. F. Chollet, *Deep Learning with Python* (Manning Publications and Co., Shelter Island, New York, 2018)
16. A. Bush, J.F. Döppler, F. Goller, G.B. Mindlin, Proc. Natl. Acad. Sci. USA **115**(33), 8436–8441 (2018)
17. F. Goller, R.A. Suthers, J. Neurophysiol. **76**(1), 287–300 (1996)
18. R. Laje, T.J. Gardner, G.B. Mindlin, Phys. Rev. E **65**(5), 051921 (2002)
19. Y.S. Perl, E.M. Arneodo, A. Amador, F. Goller, G.B. Mindlin, Phys. Rev. E **84**(5), 051909 (2011)
20. T. Gardner, G. Cecchi, M. Magnasco, R. Laje, G.B. Mindlin, Phys. Rev. Lett. **87**(20), 208101 (2001)
21. D.T. Blumstein et al., J. Appl. Ecol. **48**(3), 758–767 (2011)
22. K.H. Frommolt, K.H. Tauchert, Ecol. Inform. **21**, 4–12 (2014)
23. P.M. Stepanian, K.G. Horton, D.C. Hille, C.E. Wainwright, P.B. Chilson, J.F. Kelly, Ecol. Evol. **6**(19), 7039–7046 (2016)
24. F. Grondin, F. Michaud, Robot. Auton. Syst. **113**, 63–80 (2019)
25. S. Sturley, S. Matalonga, in *Proceedings of 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)* (2020), pp. 1–6
26. E. Hansler, G. Schmidt, *Speech and Audio Processing in Adverse Environments* (Springer, Berlin, 2008)
27. K. Miyazaki, T. Toda, T. Hayashi, K. Takeda, IEEE J. Trans. Elec. Electron. Eng. **14**(3), 340–351 (2019)
28. O. Giraudet, J.I. Mars, Appl. Acoust. **67**(11–12), 1106–1117 (2006)
29. P. Le Bot, H. Glotin, C. Gervaise, Y. Simard, in *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 1–4 (2015)
30. H. Sundar, T.V. Sreenivas, C.S. Seelamantula, IEEE/ACM Trans. Audio Speech Lang. Process. **26**(11), 1976–1990 (2018)
31. Zonotrichia capensis executing multiples themes




SPI | Journal: **11734** MS: **0405** ☐ TYPESET ☐ DISK ☐ LE ☐ CP Disp.:**2021/12/22** Pages: **10** Layout: **EPJST**

Journal: 11734
Article: 405

# Author Query Form

**Please ensure you fill out your response to the queries raised below and return this form along with your corrections**

Dear Author

During the process of typesetting your article, the following queries have arisen. Please check your typeset proof carefully against the queries listed below and mark the necessary changes either directly on the proof/online grid or in the 'Author's response' area provided below

| Query | Details required | Author's response |
|---|---|---|
| 1. | Please check and confirm if the author names and initials are correct. | |
| 2. | Please confirm if the inserted city name is correct. Amend if necessary. | |
| 3. | Please provide DOI for the references. | |
| 4. | Please update Ref. 31 with complete details. | |