Abstract—More and more animal species are endangered. In order to study and protect them, several measures have been taken, which now seem to show their limit. This work focuses on the counting process, a key issue for any project that aims to protect animals. Indeed, this paper proposes an algorithm for automatic counting of singing birds in their habitat by using wireless sensors fitted with microphone. Sensors are used to record audio samples of bird songs. These samples are used to extract some kinds of song fingerprint that are thereafter used to recognize the bird species, through a classification process. Afterwards, by analyzing data which have been collected by a central base, the network structure derived from detection fields of sensors, allows our algorithm to propose an estimate of singing birds in the habitat. This paper details the counting algorithm which uses graph theory and audio inputs comparison. Finally, we demonstrate our scheme efficiency through experimentations.

Index Terms—wireless sensors network, habitat monitoring, heuristic, performance, algorithm, simulation

I. INTRODUCTION

New challenges regarding our planet appeared in the last decades. Environmental issues that threaten our ecosystem have led scientists to work on many fundamental problems such as global warming, pollution, energy depletion, etc. One of these problems concerns the disappearance of certain animal species [23], [13]. Indeed, more and more animals are endangered mainly because of the changes which occur in their habitat. This phenomenon, often a sign of unbalanced environment, has received the attention of many scientific communities who are interested in the protection of these endangered species. An usual first step of protection or study process of species begins with the counting of individuals.

The estimate of a population density is a fundamental objective of these counting, because it enables to evaluate the population distribution but also its evolution. This information is essential to identify the prior sites for the conservation, particularly through the application of numerical thresholds on the number of present birds. Thus this measure greatly helps scientists to identify important areas for bird conservation.

However, counting individuals is not an easy task. The methods currently used are often inadequate or too expensive, and may limit researchers possibilities. Indeed, traditional methods systematically use banding of individuals [5], [30], a process that requires human presence and may generate many errors. In the case of birds, this method proves to be particularly ineffective, because birds can escape to the census process. However environmental conditions such as dense foliage or mountainous area are often unsuitable to large-scale manual counting. We can add that the human presence may generate escape behavior of birds.

Today, a solution could be provided by sensors. Technological developments and particularly the emergence of micro-devices such as wireless sensors allow thinking that new methods of study and monitoring of endangered species can be considered. Levels of performance in data processing and data storage achieved by sensors are especially useful in the case of animal counting.

The work we have conducted aims to propose an automatic solution to count songs in the bird habitat, by using sensors fitted with microphone. Indeed, birdsong is a good indicator of the population density since it enables to estimate the population size. The approach we propose is processed in two steps. To begin, the recognition process has to recognize if the recorded sound belongs with the given species. Then, the counting process analyzes the song to give an estimate of the group size.

This work was motivated by the intention of scientists to protect a bird species called “Moqueur Gorge Blanche” endemic to Martinique island. The objective is to identify areas where the species is most present and provide an estimate of the population size. Thus in this paper, we present a preliminary and essential step before deployment on the ground in order to measure the impact of all parameters on counting. The solution is studied thanks to our tool that allows simulating bird songs in a virtual environment and evaluating how much sensors are able to count correctly the songs.

This paper is organized as follows. In Section 2, we give an overview of the techniques generally used to estimate locations of multiples sources with an unknown array of sensors. We also present in this section the works on the recognition of bird species. In Section 3, the song recognition system is described in detail. Section 4 is devoted to the counting algorithm. In Section 5, the experimental results demonstrate the efficiency of the method. Finally, we conclude in Section 6 and present future directions.

Wireless Sensor Network for Habitat Monitoring: A Counting Heuristic

Erick Stattner, Nicolas Vidot, Philippe Hunel and Martine Collard
LAMIA Laboratory,
University of the French West Indies and Guiana,
FRANCE
estattne@univ-ag.fr, nvidot@martinique.univ-ag.fr, phunel@univ-ag.fr, mcollard@univ-ag.fr
II. RELATED WORK

Today, it is true that the development of large-scale distributed systems has led to new perspectives in a wide range of applications [24]. Chong et al. present in [38] sensors as “the most important technologies for the 21st century”. Martincic and Schwiebert [24] introduce the topic of sensor networks and give an overview of several application perspectives such as military, environmental or health applications. They also present a number of challenges such as location discovery, data aggregation or security. The work we conducted falls on two research areas:

A. Recognizing the species

These last years, the analysis of audio signals has been widely studied in disciplines which found applications in many fields such as speech recognition [25], [28], object detection [36] or source localization [4], [3]. The recognition of animal species has also been studied, especially for birds [9], [37] since it offers a huge advantage for automatic habitat monitoring.

Indeed, due to many analogies that exist between human speech and birdsongs [11], numerous techniques that have been designed for human speech recognition have been applied to birdsong [8], [20], [32]. For example, Fagerlund [12] proposes the use of SVM\(^1\) to identify songs of a given bird species. Similarly, Cai et al. [6] suggest the use of a neural network and Somervuo and Harma [33] propose a division into syllables. Although these methods are efficient, they cannot be directly implemented on sensors. In our previous works [14], [15], [34], [35], we have used a classification process to identify birdsong as explained in section 3. This process is detailed in the next section.

B. Source localization

Source localization is an area that has a fundamental interest in sensor network field. Indeed, in most cases, sensors are not able to locate precisely the origin of a phenomenon. Thus spatio-temporal data collection requires the development of new algorithms [7], [27]. In [29], Savvides et al. present a set of applications which require recognition of sources localization such as radars and sonar. In their work [19], Kumar and Shepherd present a military use of these techniques. They show how sensors can be used to collect, analyze and locate objects for military purposes. Valin et al. [36] design mobile robots which are able to locate objects by the sound they produce in order to avoid them. Similarly, Brandstein et al. [3] propose a complete methodology to locate the voice with a set of microphones. More generally, Krim and Viberg [17] provide an overview of main techniques used in source localization.

However, sound localization algorithms are dependent on the characteristics of the sound waves. Moreover, various parameters such as the size of bandwidth, distance to the sensor or reverberation may influence the computation of the position. Some algorithms use the time between the sensor and the source to propose an estimate of the position [31].

In our work, we do not use the acoustic properties of a bird song to find its location. We assume that all sensors know their position and we use the basic techniques of triangulation [1] to get an estimate of the birds position. Note that our algorithm does not try to precisely locate birds, but just give an estimation on the birds populating an area. Before detailing our counting system, we present in the next section how we use sensors to recognize the bird songs.

III. BIRD SPECIES RECOGNITION

The recognition of a song of a given species is the first step of the counting process we propose. Indeed, before any census system, we have to first correctly identify a given species. In our previous works [15], we proposed a recognition process that consists of two steps:

A. Parametrization process

The parametrization process allows representing audio signals by a series of coefficients that describe the signal. In this work, we use MFCC\(^2\). Indeed, different methods of parametrization exist (LPC, LPCC, PLP, etc.), but the MFCC is used here because the analysis is limited to a very small vocabulary on limited devices. In his study, Christophe Levy [22] compared different methods of parametrization on small systems, like mobile phones, for reduced vocabulary and showed that the parametrization based on MFCC is significantly more efficient for this kind of systems. After this step of parameterization, a fingerprint that represents the signal is obtained. Here, the parametrization process uses bird songs to create fingerprints. Afterwards, these fingerprints are stored into a database that gathers all recorded fingerprints of the given species.

B. Classification process

For a given song, the problem is to determine whether the associated fingerprint belongs to the species cluster or not. Inspired by the work of Rabiner and Wilpon [26], we use a classification algorithm (J48, NBTree, etc.) to determine if a fingerprint inserted to the database is characteristic of the species. If so, we consider that a bird of the species was singing and the associated data (mote location and timestamp) are stored in the database for future processing. As proposed by Kyriakopoulou and Kalambokis [21], we can also use a clustering algorithm (K-Means, EM, etc.). This technique

---

\(^1\)Support Vector Machine

\(^2\)Mel-Frequency Cepstral Coefficients
allows exploiting information derived from both training and testing sets when the training set is not sufficient. Figure 1 presents results obtained with several algorithms. We studied the recognition without noise and by adding Gaussian noise. Our recognition results are efficient since almost all birds are classified correctly.

To sum up, the counting algorithm we propose is organized into distinct tiers, as depicted on Figure 2.

1) When birds are singing, sensors sample and parametrize the song. Afterwards, they send the fingerprints obtained to the central database.

2) By using the classification process, the base station determines if the fingerprints are characteristic to the species or not.

3) The counting algorithm is run to estimate the number of birds.

The next section is devoted to the algorithm for counting we propose.

IV. THE COUNTING ALGORITHM

In our previous work [14], [15], we proposed methods inspired by the triangulation detection used by R. E. Bell [1] to count owls in the forest. These counting methods were based on research of MDS\(^3\) [14], a problem that ultimately proved to be equivalent to finding a maximal clique [2] in a UDG\(^4\) [18].

**Definition 1.** A Unit Disk Graph \(G'(V', E')\) is an intersection graph of disks of unit radius, such that \(\forall i, j \in E', \) the unit circle of center \(i\) intersects the unit circle of center \(j\). More precisely, in the case of our old approach, \(G'\) was the

---

\(^3\)Maximum Detection Set

\(^4\)Unit Disk Graph
in the UDG. We used this kind of graph because it exhibits nice properties that enable the finding of a maximal clique in polynomial time [10].

In this paper, we propose a new solution for these problems by comparing sounds recorded by sensors. We show that we no longer need to search for cliques, and consequently to use a UDG. Foremost, we assume that all the sensors form a graph of detection that we denote $G$.

**Definition 2.** A graph of detection $G$ is a graph in which two sensors $c_1$ and $c_2$ are connected, if there is an area for which a same bird can be detected by both $c_1$ and $c_2$, i.e. if there are areas where their detection fields are overlapping.

In previous works [14], [15], we considered only radius of detection, but in practice we know that the detection field of sensors are not perfectly circular. They may be influenced by obstacles, weather conditions, the quality of the device used, etc. This work takes this aspect into account. Figure 5 shows an example of a graph of detection.

Thus, the solution we propose is based on the fact that when two adjacent sensors in $G$ detect a bird song, there are two possibilities:

(i) **Either it is the same bird:** the bird was singing in the common area of the detection fields. Thus the fingerprints extracted by sensors should approximately be the same, or at least only vary very slightly.

(ii) **Or there were two different birds:** two distinct birds were singing in sensor environments. The fingerprints extracted from the songs should vary quite significantly.

In our previous versions [14], [15] of the approach we propose, the number of counted birds was equal to the number of extracted cliques in the UDG formed by sensors that detected birds. Here, we improve the method by comparing song fingerprints. In fact, the scheme we propose in this work uses fingerprints comparison only when a song of the species is detected by adjacent sensors in the detection graph. Of course, it is not necessary to compare fingerprints of songs recorded by sensors that are not neighbors in $G$, because we assume that at the same time $t$, non-adjacent sensors in $G$ could not hear the same bird. The ambiguity of cases (i) or (ii) can only occur from adjacent sensors in the detection graph.

This comparison is possible thanks to two sounds properties. First, we know that when a same bird song is detected by two sensors, the associated fingerprints will not be exactly identical but should be very close. This slight difference is essentially due to the sound environment around sensors that may vary for two adjacent sensors and minor changes that occur during signal propagation. The second element is that two birds are unlikely to sing exactly in the same way, thus when two sensors record different bird songs, the extracted fingerprints vary quite significantly. We can observe these phenomena on Figure 6. Figure 6(a) shows similarities between the fingerprints of the same bird song recorded by two adjacent sensors in $G$. Similarly, Figure 6(b) shows wide variations that exist between fingerprints of two different bird songs recorded by adjacent sensors.

If we assume sensors are able to correctly differentiate bird songs, it would be possible to realize an estimate of the number of bird songs by using only this property. However, the comparison may sometimes fail because of the ambient noise. Let us take the case of two sensors. One located close to a watercourse, and the other in a non-disturbed environment. If these sensors are adjacent in $G$ and hear the song from the same bird, it is possible that the fingerprint comparison fails. So it is easy to understand that the probability that such errors occur, increases when the number of comparisons increases too. Thus it is necessary to reduce the number of fingerprint comparisons to a bare minimum.

In our context, we assume that all sensors have the same capabilities and the same lifespan. They are able to record songs and send associated fingerprints to the central base $B$. The central base receives data from all the sensors in the given area. In addition, we do not have any information on the number of birds, their movements or their behaviors. We are not interested here in the routing of information in the sensor network. We only focus on bird counting.
More formally, let denote \( C = \{c_1, ..., c_n\} \) the set of sensors which cover the study area, \( E \) the set of edges between sensors, and \( G(C, E) \) the detection graph formed by overlapping of detection fields of sensors of the set \( C \).

More precisely, the detection graph \( G(C, E) \) is the graph of sensors formed by \( C \) such that \( \forall i, j \in C \), if the detection fields of \( i \) and \( j \) are overlapping, \( ij \in E \). So there exists an area for which \( i \) and \( j \) can record the song of the same bird. So \( E \) is the set of sensor pairwise that share common areas in their detection fields.

Similarly, let denote \( P \) the set of song fingerprints that can be detected and \( S_e \) the threshold from which two fingerprints are sufficiently close to correspond to the same song. We denote \( F_t \) the detection function that returns for a sensor \( c \) the fingerprint of the song detected by \( c \) at a time \( t \), if it corresponds to a song of the targeted species.

\[
F_t : C \rightarrow P
\]  

We assume that if several birds are singing near a sensor \( c \), they generate noise in the environment of \( c \). This sensor cannot detect them. Moreover, at a time \( t \), a sensor may detect no bird. The base station maintains the detection vector \( \overrightarrow{D}_t \) that represents the state of detection of each sensor at time \( t \).

\[
\overrightarrow{D}_t = [\phi_t(c_1), ..., \phi_t(c_n)] \quad \text{with} \quad \forall i \in [1..n], \quad \phi_t(c_i) = \begin{cases} F_t(c_i) & \text{if } c_i \text{ detected a song at } t \\ \emptyset & \text{otherwise} \end{cases}
\]  

To estimate the number of birds that are singing at time \( t \), we start by building from \( \overrightarrow{D}_t \), the set \( C'_t \), with \( C'_t \subset C \), only composed of sensors that detected a song of the species at time \( t \). i.e.

\[
\forall c_i \in C, \quad c_i \in C'_t \quad \text{if and only if} \quad \phi_t(c_i) \neq \emptyset
\]  

We call \( C'_t \) the set of detection. Thus the sub-graph of \( G(C, E) \) formed by the sensors of \( C'_t \) is not necessarily connected. Once \( C'_t \) is obtained, we seek for each sensor \( c_i \) of \( C'_t \), the set \( S_t(c_i) \) of neighbors \( c_j \) such as:

\[
\forall c_i \in C'_t, \quad S_t(c_i) = \{c_j \in C'_t / c_i, c_j \in E \text{ and } |F_t(c_i) - F_t(c_j)| < S_e\}
\]

Then, we remove \( c_i \) and all sensors of the set \( S_t(c_i) \) from \( C'_t \) and we increment the number of birds. This operation is repeated until \( C'_t \) does not contains any sensors i.e. \( C'_t = \emptyset \).

The comparison of the fingerprints is processed by computing the distance between the two song fingerprints [16]. The basic principle is to evaluate the similarity between the two bird songs recorded by evaluating the distance that separates them.

\[ \text{Algorithm 1 Heuristic for Counting of Individuals} \]

**Require:** \( D = \overrightarrow{D}_1, ..., \overrightarrow{D}_m \): A list of detection vectors

1. \( NbBirds \leftarrow 0 \): Estimation of the number of individuals
2. \( L \leftarrow \emptyset \): List
3. for \( k \) from 1 to \( m \) do
4. \( NbBirds \leftarrow 0 \)
5. Create \( C'_k \) from \( \overrightarrow{D}_k \)
6. while \( C'_k \neq \emptyset \) do
7. \( i \leftarrow \text{first sensor of } C'_k \)
8. Generate \( S_t(i) \)
9. Remove \( i \) and all sensors of \( S_t(i) \) from \( C'_k \)
10. \( NbBirds \leftarrow NbBirds + 1 \)
11. end while
12. add \( NbBirds \) to \( L \)
13. end for
14. return \( \max_{l \in L} l \)

Indeed, before deploying our solution on the ground, it appears necessary to measure all phenomena that may influence the method. Consequently, simulation was a first step before the real deployment of the wireless sensors and is expected to reveal relevant information for optimizing the deployment of sensors in the bird habitat. Simulations have indeed become essential steps in the study and understanding of complex phenomena. This section is devoted to the simulation tool and results.

To assess the efficiency of the algorithm, we developed a 2D simulation environment in Java that simulates birds in their habitat. The application aims to provide a simulated environment, which is able to virtually reproduce bird habitat in which we place the virtual network of sensors.

The tool allows us to generate detection fields with random shape for sensors. The base station knows the detection graph formed by the sensors. Although in practice, obtaining the real detection graph is not always possible, it is possible to generate a theoretical detection graph that simply corresponds to the range of sensor microphones. However, the objective here is to evaluate the efficiency of the method in conditions close to reality.

We assume that communications with the central base are always successful. Indeed, we are not interested in the optimization of routes neither to the base station nor to any loss or corruptions of information. Sensors are still able to communicate with the base station. As in real conditions, we do not have any information about the movements of birds. Indeed, the knowledge that we have of these birds is very limited. Also, in order to remain close to reality we did not give any particular behavior to the birds. In our simulation birds can fly at 2 meters per second, stay at a place, take random directions or sing with a certain probability defined during the calibration of the application.

The tool is fully customizable. Thus, it is possible to define the size of the study area, the number of sensors, their detection field, the number of birds, the probability that birds

V. Experimental Results

In order to apply the counting method in real conditions at medium term on the site of “La Caravelle”, the protected habitat of the “Moqueurs Gorge Blanche”, a first necessary step was to observe the impact of all factors implicated on results.
sing, the movement speed, etc.

Figure 7 shows the visualization interface of the 2D simulation. The large rectangle represents the study area. Sensors are represented by filled circle and we can observe the detection graph that binds them. Birds that are singing are represented by filled squares. You can notice that the graph of detection is not always connected, as in a real context.

We implemented the algorithm presented in the previous section on the simulation tool. The results were obtained with a calibration that we consider close to the real environment. The dimension of the study area is about 1000m × 1000m. Sensors are set randomly on the study area and birds sing with a probability of 20%. As said previously, we assume that when several birds are singing near a same sensor, they generate noise in the environment of the sensor which has the effect of disturbing the recognition process. Thus these birds are not detected. The results were obtained with the simulation environment as follows:

- Intel Core 2 Duo P8600 2.4 Ghz, 4Go RAM
- Linux Ubuntu 9.04, JDK 1.6

Each test was performed on a base of 100 executions. Then, we calculated the average of the results obtained to measure the efficiency of the algorithm. In the tool, the sensors are placed randomly on the area, a placement close to what is obtained by dropping sensors from plane. So we can suppose that the study area will not be covered entirely, especially in the case of wide area. For instance, Figure 7 shows 60 sensors set randomly on an area of 450m × 250m. We can observe that the study area is not fully covered. Some birds may escape to the counting process. That is why we propose two measures. First, we define \( \text{R1} \) as the evaluation of the algorithm depending on the real number of birds, i.e. all birds are considered including those that were singing on not-covered areas. Second, we define \( \text{R2} \), as the evaluation of birds singing on covered areas, i.e. “those we should have count”.

Figure 8 presents the counting rate of the algorithm with different network sizes and Figure 10 presents the counting rate for different population sizes. In the tables:

- \( \text{NbS} \) is the number of sensors,
- \( \text{NbB} \) is the number of birds,
- \( \text{NbR} \) is the number of real birds singing,
- \( \text{Algo} \) is the number of birds counted by the algorithm,
- \( \text{R1} \) is the counting rate depending on \( \text{NbR} \),
- \( \text{R2} \) is the counting rate depending on \( \text{NbC} \).

Figure 8 shows the estimate of the counting rate depending on the network size. We can notice that the method presents good performance for birds that are singing in covered areas. Indeed, the number of counted birds is very close to the number of birds that are singing in covered areas. However, the study area is large and when the sensors number is not sufficient to cover the whole area, many birds escape to the counting process. This is why bad results are observed for \( \text{R1} \) when the number of sensors is not sufficient. When the number of sensors increases, the surface coverage becomes more important and the number of counted birds is close to the real number of birds.

The under-counting phenomenon is due to the case of several birds singing in sensor environments. They generate noise and cancel the recognition of the songs. However, the addition of new sensors can reduce this phenomenon. Let us
<table>
<thead>
<tr>
<th>NbB</th>
<th>NbR</th>
<th>NbC</th>
<th>Algo</th>
<th>R1(%)</th>
<th>R2(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2,00</td>
<td>1,90</td>
<td>1,89</td>
<td>94,21</td>
<td>99,16</td>
</tr>
<tr>
<td>20</td>
<td>4,01</td>
<td>3,81</td>
<td>3,76</td>
<td>93,81</td>
<td>98,74</td>
</tr>
<tr>
<td>30</td>
<td>5,83</td>
<td>5,50</td>
<td>5,39</td>
<td>92,39</td>
<td>97,96</td>
</tr>
<tr>
<td>40</td>
<td>7,94</td>
<td>7,52</td>
<td>7,27</td>
<td>91,63</td>
<td>96,70</td>
</tr>
<tr>
<td>50</td>
<td>10,13</td>
<td>9,60</td>
<td>9,24</td>
<td>91,24</td>
<td>96,33</td>
</tr>
<tr>
<td>60</td>
<td>11,96</td>
<td>11,38</td>
<td>10,82</td>
<td>90,44</td>
<td>95,08</td>
</tr>
<tr>
<td>70</td>
<td>13,94</td>
<td>13,18</td>
<td>12,46</td>
<td>89,33</td>
<td>94,51</td>
</tr>
<tr>
<td>80</td>
<td>15,92</td>
<td>15,05</td>
<td>14,14</td>
<td>88,84</td>
<td>93,99</td>
</tr>
<tr>
<td>90</td>
<td>17,86</td>
<td>16,87</td>
<td>15,66</td>
<td>87,64</td>
<td>92,81</td>
</tr>
<tr>
<td>100</td>
<td>19,81</td>
<td>18,76</td>
<td>17,16</td>
<td>86,63</td>
<td>91,51</td>
</tr>
</tbody>
</table>

Fig. 10. The Impact of the Population Size with a Configuration of 1000 Sensors

The results for the counting rate are as follows:

- **R1**: The counting rate is above 90% for all population sizes, with a slight decrease as the population size increases. This suggests that the method is robust against increases in population density.
- **R2**: Similar to R1, the counting rate is also above 90%, indicating good performance across different population sizes.

These results highlight the dependency of the method on the number of sensors and the population size. With a constant number of sensors, the counting rate decreases as the population size increases. However, even with a high population size, the method maintains a high success rate, making it suitable for large-scale bird monitoring.

**VI. Conclusion and Future Work**

There are more and more endangered animal species. This problem, which has received the attention of many scientific communities, could find a solution through sensor networks. Indeed, sensors seem to offer real opportunities for study, monitoring, and modeling the habitat of these animals.

In this paper, we explore one of these possibilities. We have proposed a network of sensors fitted with microphones to estimate the number of birds in their habitat. Unlike our previous work, we show that our method is more suited to a real environment. Indeed, the method we propose could be used with sensors, whose detection area is not perfectly circular. Our solution that is based on the exploitation of a detection graph and comparison of audio signals has proved to give good results when the number of sensors is high.

However, we can also notice that the sensor network size is not the only factor responsible for errors. Indeed, for large bird populations, we noticed that the performance of the counting process decreases even when the network size is high, as shown on Figure 10. This is because we cannot obtain a configuration in which one sensor catches one song and only one. Indeed, when the population size increases, the probability that birds sing in close proximity to each other increases too.

It would be interesting to see how the algorithm behaves according to these two parameters. To measure the impact of these phenomena, we represent the performance by 3D graphics. Figure 11 presents the results obtained. Overall, we can observe that when the number of sensors is greater than 500, the method provides good results. However, as observed on Figure 10, we notice that even for large networks, there exist slight variations in performance when the population size is varying.

We showed how the addition of sensors could reduce the noise effect observed in the sensors environment as shown on Figure 9. Thus with a network of 2000 sensors and a population of 100 birds, we obtained a success rate above 90% for R1 and R2.
to adjust the placement of sensors in order to achieve optimal counting.

These first results confirm the opportunity to set up the network on the ground in the real environment of the Moqueur Gorge Blanche. However, this work only constitutes a first stage in a wide project whose goal is to use sensor networks for gathering information and improving knowledge we have about endangered animals. In the short term we will focus on the adaptation and improvement of the solution for the deployment on the ground. Moreover, in the longer term we want to use sensors for collect at regular time intervals, data such as temperature, light or noise to understand habitat of the birds. Afterwards, it would be interesting to use this information to replicate the habitat of these animals.

REFERENCES


