Vol.2009-MPS-75 No.26
2009/9/11

Automatic Detection and Recognition of the Behavioral Sequences of Bengalese Finch Song by Using Image Processing

Khan Md. Mahfuzus Salam,[†1] Tetsuro Nishino,[†1]
Kazutoshi Sasahara,[†2] Miki Takahasi[†2]
and Kazuo Okanoya[†2]

The Bengalese finch song has been widely studied for its unique features and similarity to human language. For computational analysis of the songs, they must first be represented in terms of behavioral sequences. This paper introduces a new approach for automatic detection and reorganization of the behavioral sequences via image processing. This approach is based on the recognition process used by a human to visually identify patterns in a spectrogram. The behavioral elements of birdsong are independent to birds (i.e, similar pattern does not appear in two birds). Considering this constraint, we believe that the proposed method is a generalized approach. On real birdsong data, we find that our method achieves a high accuracy rate. Thus, we consider our method a feasible approach.

1. Introduction

Birdsong has been actively studied via analysis of behavioral data sequences to understand the language model of birds. The songs of the Bengalese finch (*Lonchura striata var. domestica*), which is a popular fowl in Japan, is widely studied for this purpose. The song of the Bengalese finch has a complex structure as compared with those of other songbirds such as zebra finches (*Taeniopygia guttata*). According to the recent studies, the courtship songs of Bengalese finches have unique features and similarity with a human language 5). Thus, Bengalese finch songs have been studied as a model of human language. In birdsong research, an acoustic song analysis is necessary to find the song elements and their sequence for carrying out an analysis to understand the song syntax 8) and the learning process of the song. The current research is focused on automatic detection and recognition of the song elements. Previous studies that employed sound processing had drawbacks. This paper introduces a new generalized approach that employs image processing to overcome the drawbacks.

†1 Information and Communication Engineering, The University of Electro-Communications, Tokyo, Japan.
†2 Laboratory for Biolinguistics, RIKEN Brain Science Institute (BSI), Saitama, Japan.

2. Preliminaries

This section briefly introduces the theoretical foundations of a birdsong, its representations, image basics, and the recognition process by humans as we focused on the recognition process that is manually carried out by humans.

2.1 Bengalese finch song

Recent studies on Bengalese finches show that the songs of male Bengalese finches are neither monotonous nor random; they consist of chunks, each of which is a fixed sequence of a few song notes. The song of each individual can be represented by a finite automaton, which is called song syntax (**Fig. 2**) 5). Thus, the songs of Bengalese finches have *double articulation*, which is one of the important faculties of human language (i.e., a sentence consists of words, and each word consists of phonemes). Because of the structural and functional similarities of vocal leaning between songbirds and humans, the former have been actively studied as a good model of a human language 3). In particular, the song syntax of Bengalese finches sheds light on the biological foundations of syntax.

2.2 Birdsong representation

In birdsong analysis, the song data is recorded in an appropriate environment. From the recorded sound data, we obtain the spectrogram of the song. For further computational analysis, a spectrogram is used as the standard representation of the song. The later part of this section briefly explains some general terms that are used in birdsong research.

Behavioral element: An independent pattern that appears in a spectrogram and is assigned a symbol is called a song element or song note or behavioral element (**Fig. 1**). In this paper, we use the term behavioral element. From the definition, we can say the text data that consist of symbols like a, b, c, and d are called behavioral sequences. Song notes correspond to phonemes of human languages.

Spectrogram: A spectrogram is an image that shows how the spectral density of a signal varies with time. It is also known as a sonogram. Spectrograms are used to identify phonetic sounds and to analyze the cries of animals, speech processing, seismology, etc.

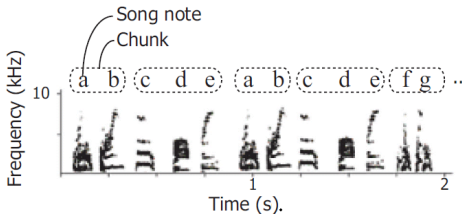


Fig. 1 Grayscale spectrogram of a Bengalese finch song

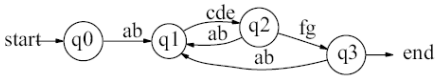


Fig. 2 Courtship song syntax represented by an automata

There are many variations in the format of the spectrogram. Sometimes, the vertical and horizontal axes are switched; sometimes, the amplitude is represented as the height of a 3D surface instead of color or intensity. The most common format that is used in birdsong research is a graph with two geometric dimensions: the horizontal axis represents time, and the vertical axis is frequency; a third dimension indicating the amplitude of a particular frequency at a particular time is represented by the intensity or color of each point in the image. Fig. 1 shows a sample grayscale spectrogram of a Bengalese finch courtship song.

2.3 Detection and recognition

Human vision is one of the most important and perceptive mechanisms. It provides information required for the relatively simple tasks (e.g., object recognition) and for very complex tasks as well. In bird song research, the behavioral element recognition is carried out by humans by inspecting the patterns visually represented in a spectrogram.

2.3.1 Image feature extraction

Digital image processing based on property denotes the analysis carried out on the basis of the pixel property of the image irrespective of the image type. A digital image has a finite set of digital values called picture elements or pixels. The image contains a fixed number of rows and columns of pixels. Each pixel of a raster image is typically associated with a specific *position* in some 2D region and has a value of one or more quantities related to that position. Digital images can be classified according to the number and nature of such samples into the different categories like *Binary, Grayscale, Color, False-color*. In our research, we use a spectrogram that is a grayscale image.

Grayscale Image: A grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Grayscale images have many shades of gray in between. The reason for differentiating such images from any other sort of color image is that less information needs to be provided for each pixel. In fact a *gray* color is one in which the red, green, and blue components all have equal intensity in the RGB space, and hence, it is only necessary to specify a single intensity value for each pixel, as opposed to the three intensities needed to specify each pixel in a full color image. Grayscale images are also called monochromatic images, denoting the absence of any chromatic variation.

Pixel Values: Each of the pixels that represent an image stored inside a computer has a pixel value that describes how bright that pixel is, and/or what color it should be. For a grayscale image, the pixel value is a single number that represents the brightness of the pixel. Presently, grayscale images are commonly stored with 8 bits per sampled pixel, which allows 256 different intensities (i.e., shades of gray). The precision provided by this format is barely sufficient to avoid visible banding artifacts but is very convenient for programming because a single pixel occupies less space than a single byte. The binary representations assume that 0 is black and the maximum value 255 is white. In our research, we use the grayscale spectrogram images that have pixel intensity values from 0 to 255.

2.3.2 Image matching and pattern recognition

Pattern recognition aims to classify data or patterns on the basis of either a priori knowledge or statistical information extracted from the patterns. The patterns to be classified are usually groups of measurements or observations, defining points in an appropriate multidimensional space. This is in contrast to pattern matching, where the pattern is rigidly specified. Pattern recognition is used to test whether things have a desired structure, to find relevant structure, to

retrieve the aligning parts, and to substitute the matching part with something else.

Current research applies image processing based on grayscale image features. The motivation of applying such image processing is to find a simple and generalized way for the automation as a human brain does in the recognition process by applying pattern matching.

3. Methodology

The proposed automation process is divided into two steps. First, from the song spectrogram, we detect the behavioral elements on the basis of the local property of the spectrogram image. Then, on the basis of the detected elements, we apply image matching to assign a label to the extracted elements, and thus, we obtain the behavioral sequence of the song.

3.1 Behavioral element detection

From the spectrogram, we first detect the elements. On the basis of the extracted statistical features of the detected elements, we carry out the recognition process. For this reason, the detection process is very important.

3.1.1 Detection method

The detection process is carried out by analyzing the spectrogram for intensity values; we can obtain a graph for the average pixel intensity value. If the spectrogram has many noises at the beginning, which are ignored in the visual inspection by human, the present system does not ignore them as noises. For this reason, we preprocess the spectrogram. Then, if we take the average intensity value along the vertical line and draw a graph where the Y-axis represents the average intensity value or gray value and the X-axis represents the pixel index x, which is the distance from the (0, 0) pixel along the X-axis, we have a graph that is shown in **Fig. 5**. It is clearly visible that from the graph we can find some clear gaps between the elements. By defining parameters such as *minimum element width, minimum gap between elements, and the intensity threshold*, we can execute our algorithm to find the song elements. If some region does not fit with the three above mentioned parameters, we consider it to be noise. Note that these parameters can vary from bird to bird. The detected behavioral elements and the features of the elements, such as width information, are used for the recognition process.

3.1.2 Detection algorithm

The behavioral element detection algorithm takes the array of the average intensity values as the input. On the basis of the defined parameter values, the proposed detection algorithm produces an unlabelled list of behavioral elements.

Detection Algorithm

Input: array of intensity values.
Output: a list of elements.
Procedure:
(1) Initialize the parameters.
(2) If the intensity value exceeds threshold and next is not a gap
  · set start element flag true;
  · set start index to current index;
(3) If start element flag is true and next minimum gap is detected
  · set start element flag false;
  · set end index to current index;
  · add to element list;
(4) Continue step 2 and 3 until end of the intensity array
(5) Return element list.

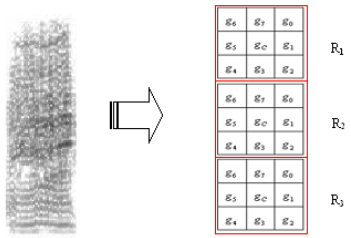
3.2 Behavioral element recognition

For extracting the behavioral sequences from the spectrogram, we extract local statistical features and then carry out the statistical pattern matching for recognition.

3.2.1 Recognition method

As discussed in the previous section, similar patterns are assigned with the same label in the recognition process. Our recognition method is based on the local property of the spectrogram. By executing the note detection algorithm, we obtain element list information. This unlabeled element list provides the start pixel and the end pixel information for every element.

As for the Bengalese finch song, note patterns differ from bird to bird. Therefore, we decided not to use any prior knowledge; rather, we use the statistical information extracted from the patterns. First, we divide every note into N regions, and every region is divided into nine (3×3) cells. We denote the center



$R_1 \cup R_2 \cup R_3 = \{R_1g_0, \ldots, R_1g_7, R_1g_c, R_2g_0, \ldots, R_2g_7, R_2g_c, R_3g_0, \ldots, R_3g_7, R_3g_c\}$

Fig. 3 Explains the feature extraction procedure while $N = 3$

cell as g_c and the other cells as g_n in a clockwise direction, where $n = 0, 1, \ldots, 7$. Thus, we obtain a set of values for every single element. Then, we apply a statistical test called the chi-square test to find the similarity between elements. Note that the value of N should not greater than 3 because if the set size exceeds thirty, the Chi-square distribution tends toward a normal distribution.

3.2.2 Chi-square goodness fit test

The chi-square test (χ^2) is a statistical hypothesis test whose results are evaluated by reference to the chi-square distribution. Pearsons chi-square test is the best-known of several chi-square tests. Its properties were first investigated by Karl Pearson 7), it is the original and most widely-used chi-square test.

When an analyst attempts to fit a statistical model to observed data, he or she may wonder how well the model actually reflects the data. How *close* are the observed values to those that would be expected under the fitted model? One statistical test that addresses this issue is the chi-square goodness of fit test. The chi-square statistic is calculated by finding the difference between each observed and theoretical frequency for each possible outcome, squaring these values, dividing each by the theoretical frequency, and taking the sum of the results. In the equation, χ^2 is of the form:

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Where,

χ^2 = the test statistic that asymptotically approaches a χ^2 distribution;

O_i = an observed frequency;

E_i = an expected frequency, asserted by the null hypothesis;

n = the number of possible outcomes;

The number of degrees of freedom is equal to the number of possible outcomes minus 1. If the computed test statistic is larger than the chi-square table 7) with (n–1) degrees of freedom, the observed and expected values are not close.

3.2.3 Recognition algorithm

The behavioral element recognition algorithm takes the unlabelled list of behavioral elements. It applies the goodness of fit test to find the similarity between elements and produces the behavioral sequence.

Recognition Algorithm

Input: unlabeled list of elements.
Output: labeled list of elements.
Procedure:
(1) For each element in element list divide into $N \times 9$ cells where $0 < N < 4$
(2) Calculate the average intensity value for every cells
(3) For each element until there is any unlabeled element
  · set one as expected and others as observed;
  · if expected is not labeled set it with a new label;
  · test the Chi-square statistics;
(4) If the observed element pass the test then set the element with same label
(5) Return updated element list

4. Results

In this section, we present the results of our methodology for analyzing the Bengalese finch song. First, we explain the nature of our real song data, and then discuss the results of the automatic detection and recognition of the behavioral sequence.

4.1 Description of data

We use two Bengalese finch song; the names of the finches are *Hikari 52* and *Hikari 49*. The song data were recorded at the *Okanoya laboratory* of RIKEN.

For *Hikari 52*, the sample spectrogram has forty six notes, and for *Hikari 49*, the sample spectrogram has fifty one notes. Both sets of sample data are part

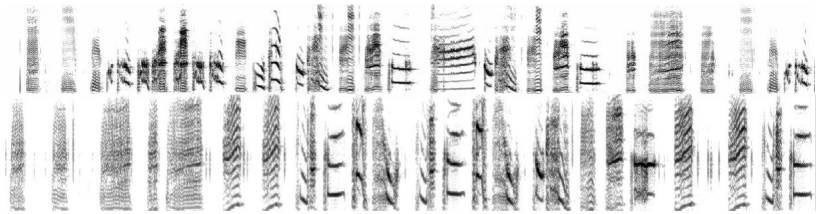


Fig. 4 Spectrogram for *Hikari 49* (top) and *Hikari 52* (bottom)

of one song. **Fig. 4** show the partial spectrograms for the two birds. From the above sample spectrogram, it is clearly visible that the spectrogram of *Hikari 49* is more complex than that of *Hikari 52*. For *Hikari 52*, the song notes are almost clearly separated from one another, but for *Hikari 49*, the song notes are not clearly separated from one another.

By applying our methodology, we implemented an application in java, which takes the spectrogram as the input and provides the extracted behavioral elements and their sequence as the output. *ImageJ API* 6) is used for analyzing the image property.

4.2 Behavioral element detection

In section 3.1, we discussed the song note extraction methodology and explained the algorithm used for extracting the song notes from a spectrogram. We used parameters such as minimum note width, intensity threshold, and minimum gap between notes. We set the parameter values for minimum note width as 10 pixels, intensity threshold as 250, and minimum gap between notes as 5 pixels for both birds. After executing the algorithm mentioned in section 3.1.2, we obtain the result for the best case as follows:

Table 1 Results of the automatic detection of behavioral elements

Bird name	Appeared element	Extracted element	Accuracy rate
Hikari 52	46	46	100%
Hikari 49	51	45	90%

Now in the case of *Hikari 52*, when we inspect the extracted patterns, we find that there are some noises with the extracted patterns although we have a good accuracy rate. To avoid the noise, if we apply a cutoff level of 30 at the intensity value graph, we obtain 40 extracted elements. Therefore, the accuracy rate decreases, and certain elements lose some necessary information, which is not desirable. Fig. 5 describes the noise situation.

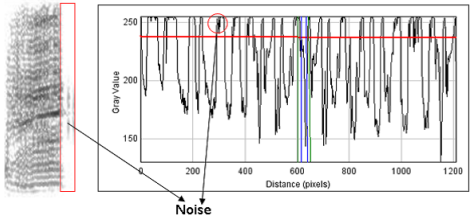


Fig. 5 Description of noise and effect of applying cutoff level for *Hikari 52*

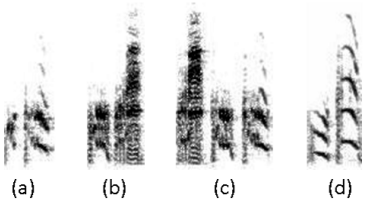


Fig. 6 Description of the error in the detection for *Hikari 49*

In the case of Hikari 49, when we inspect the extracted patterns, we find that some song notes are not extracted correctly. Initially, we have an accuracy rate of 75% with our default parameter value as the gaps between the elements are too short to separate. Fig. 5 describes the errors in the detection process.

Fig. 6, except Fig. 6(d), shows some incorrect extracted notes for *Hikari 49*. If we carefully inspect Fig. 6, we can observe that Fig. 6(a) and Fig. 6(b) should be extracted as two different elements because the right pattern in figure Fig. 6(a) and Fig. 6(b) is appears separately (see, Fig. 6(c)) in the spectrogram, and Fig. 6(c) should be extracted as three different elements However, Fig. 6(d) is considered to be extracted as a right pattern although it has the same nature as the patterns shown in Fig. 6(a, b, and c) because the two patterns are very close and the left and the right patterns do not appear separately in the song. We adjust the default parameter value of the minimum gap between the notes to be two pixels and use the cutoff level of nine. Thus, we obtain the best case result with an accuracy rate of 90

4.3 Automatic recognition

In section 3.2, we discussed the behavioral element recognition methodology and explained the algorithm. The first step is to divide every extracted element into N parts, and then calculate the average intensity value for every region. Thus, for every element, we have a set of 27 element while N = 3. Then, we ap-

ply the Chi-square test considering the note width information. In the proposed method, we compare the elements if the note width is greater than three-fourths or smaller than five-fourths of the observed element. After executing the algorithm mentioned in section 3.2.3, we obtain the behavioral sequences.

Our system produces the behavioral sequence for *Hikari 52* and *Hikari 49* as follows:

System (*Hikari 52*):
*AAB***AC***DDEFGHEFGHIBJK***L***DEFAAB***AC***DDEFGHEFGHI***B***JK***L***DEF*
Correct (*Hikari 52*):
*AAB***LB***DDEFGHEFGHICJK***D***DEFAAB***LB***DDEFGHEFGHI***C***JK***D***DEF*

System (*Hikari 49*):
*ABC***D***EFGHIJKDLIJKDMNOBCP***D***EQDGH***R***JK***S***LIJ***TS***MNABC*
Correct (*Hikari 49*):
*ABC***S***EFGHIJKDLIJKDMNOBCP***S***EQDGH***I***JK***D***LIJ***KD***MNABC*

We can summarize the result for the recognition as follows:

Table 2 Results of the song note recognition

Bird name	Accuracy rate
Hikari 52	86%
Hikari 49	85%

Notice that for *Hikari 49*, the result is based on extracted patterns in the previous step. If we consider the wrong extracted pattern, then the accuracy rate become around 70%.

If we inspect the wrong decisions made by the system for *Hikari 52*, we find that note *B* is labeled as *C* and note *L* is labeled as *D*. This is because the incorrectly labeled note contains a considerable noise (white part), which affects the matching process. In the case of incorrectly labeling note *L* note *A* for *Hikari 52*, by carefully observing each note, we find that the intensity density is the same for both the notes. This causes the recognition error and is a limitation of the proposed image matching algorithm. We notice a similar recognition error in the case of *Hikari 49*.

5. Conclusion and Discussion

Compared with previous approaches based on sound processing, the present study proposes a new approach to automatic recognition of behavioral sequences, and by applying image processing, we obtain good results for the approach. The main advantage of the proposed approach is its simplicity and feasibility. The approach is focused on a generalized (does not depend on the type of bird) process. Further the accuracy rate of the proposed approach is similar to that of other methods such as sound processing. However, sound processing requires considerable human effort (and is time consuming) for fixing the parameter values or training the system for detecting and recognizing the behavioral elements. This is not practical for an automated system. In contrast, the proposed methodology is almost automated and feasible for songbirds as our approach represents the human inspection method and does not depend on birds. The accuracy rate is also satisfactorily high. Thus, our approach saves time and is practical as an automated system.

References

1) C. K. Catchpole and P. J. B. Slater: *Bird Song: Biological Themes and Variations* , Cambridge University Press, 2nd edition (2003).
2) E. Honda and K. Okanoya: Acoustical and Syntactical Comparisons between Songs of the White-backed Munia (Lonchura striata) and Its Domesticated Strain, the Bengalese Finch (Lonchura striata var. domestica), *Zoological Science*, Vol.16, pp. 319–326 (1999).
3) J. Doupe and P. K. Kuhl: Birdsong and Human Speech: Common Themes and Mechanisms, *Annual Reviews Neuroscience*, Vol.22, pp.567–631 (1999).
4) J. Nishikawa and K. Okanoya: Dynamical Neural Representation of Song Syntax in Bengalese Finch: a Model Study, *Ornithological Science*, pp.95–103 (2006).
5) K. Okanoya: Song Syntax in Bengalese Finches: Proximate and Ultimate Analyses, *Advances in the Study of Behavior*, Vol.34, pp.297–346 (2004).
6) National Institutes of Health, USA. *ImageJ 1.41*, URL: http://rsbweb.nih.gov/ij/; last accessed: July 31st, 2009.
7) Sheldon M. Ross: *Introduction to Probability and Statistics for Engineers and Scientists*, Elsevier Academic Press, 3rd edition (2004).
8) Y. Kakishita, K. Sasahara, T. Nishino, M. Takahasi and K. Okanoya: Ethological Data Mining: an Automata-Based Approach to Extract Behavioral Units and Rules, *Data Mining and Knowledge Discovery*, Vol.18, No.3, pp.446–471 (2009).