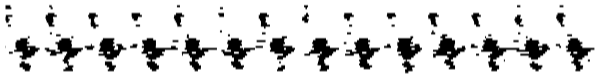
# Detecting Bird Sounds via Periodic Structures: A Robust Pattern Recognition Approach to Unsupervised Animal Monitoring

**Diplomarbeit**

von
**Daniel Wolff**



Rheinische Friedrich–Wilhelms–Universität Bonn
Institut für Informatik III

May 28, 2008

**Wherever we are, what we hear is mostly noise. When we ignore it, it disturbs us. When we listen to it, we find it fascinating.**

– *John Cage*

# Acknowledgements

Während der Anfertigung der vorliegenden Arbeit, sowie durch das vorbereitende Studium hindurch, haben mich viele Personen unterstützt und damit auch einen wesentlichen Teil zum Abschluss dieser Arbeit beigetragen. Hier gilt es vor Allem, einen Dank vorauszuschicken an diejenigen Menschen, welche mir, obwohl ich sie während dieser Zeit vernachlässigt habe, zu entscheidenden Zeitpunkten dennoch nahe gestanden haben. Insbesondere bin ich den Folgenden zum Dank verpflichtet:

Die gesamte Arbeitsgruppe von *Michael Clausen*, in der diese Diplomarbeit entstanden ist, stand mir immer hilf- und lehrreich zur Seite. Hier gilt mein besonderer Dank *Frank Kurth*, *Rolf Bardeli* und *Christian Fremerey*, welche mir auch über die Entstehung der Arbeit hinaus in vieler Hinsicht geholfen haben.

Meiner Familie gilt ebenfalls ein besonderer Dank, da mich gerade die liebevolle und vielseitige Unterstützung meiner Eltern sicher durch mein Studium geführt hat.

Mein abschließender, aber nicht minderer Dank gilt den Freunden *Kevin Kozinna*, *Fabian Liboschik* und *Kathrin Scheps* für ihre Unterstützung und Ermutigung, die mir die Anfertigung dieser Arbeit sehr erleichtert hat.

# Contents

# Chapter 1

# Introduction

Birds, though being quite common even in our biggest cities, are threatened by the impact of mankind like many other animals. As the modern culture has become aware of its influence on the surrounding ecological system, several measures are being elaborated to reduce the adverse effects of growing cities and industrial parks. In order to provide a basis for a precise and effective application of such measures, like the definition of nature protection areas, data sets on the actual inhabitants of distinct geographic zones are of great importance.

For the case of songbirds, the usual approaches to the determination of the sizes of their populations depend on the manpower available to the executive organisation. In Germany, the DDA (Association of Avifaunists) conducts such a project by line mapping a subset of 1000 watching sites [MSHRD05]. For each site, the actual observation is performed by hobby ornithologists. Here, the gathered data is subject to the perception of the respective observer: the hearing and visual capabilities as well as the expertise of each ornithologist influences the actual number of detected birdsongs. Furthermore, the observers are limited in their observation time.

Thus, in the past decade, as modern computers provide the means to analyse the growing data sets being collected at long-time acoustic recordings, the challenge to suspend or, in some cases, to substitute the human ornithologist by a machine, has been faced. Although the expertise of an ornithologist is usually not outperformed when handling a small set of acoustic data, computational methods provide powerful tools for the analysis of large databases. For example, many software tools for bioacoustic audio analysis as XBAT[XBA], Avisoft SaSLab[SAS] or DSProlog provide an automatic search for template excerpts being defined by the users before. Thus, the scientist may check the reduced data set of found matches. Recently, a query-by-example interface has been developed for the Animal Sound Archive in Berlin [TSA], allowing the user to search for recordings containing animal sounds similar to a provided query recording. Actually, the majority of recordings used for the evaluation of the proposed methods were drawn from the above archive and a set of monitoring recordings being provided in a cooperative project [FT07], respectively.

Embedded in this project, which has been founded by the German National Agency for Nature Conservation, the techniques proposed in this work aim at providing tools for the automatic detection of bird songs in real-world monitoring scenarios. Notably, most of the research performed to this point is based on more or less clean recordings of animal sounds. Dealing with a massively open source of acoustic data, featuring a wide spectrum of interfering noise as well as a less controlled set of species and individuals, this work focusses on robust feature extraction methods. Thus, two sets of features are introduced: the first set concentrates on a robust representation of the spectral, or tonal, characteristics of birdsongs. Exploiting the periodic character of many birdsongs, the features in the second set robustly derive the parameters of the typical element repetitions.

Furthermore, two detectors are presented envisaging a robust and generic recognition of two bird species: the Chaffinch and Savi's Warbler. The songs of both of the mentioned species reveal a repetitive structure, providing a basis for the deployment of features analysing the parameters of such repetitions. Both, the feature extraction routines and detectors, were implemented in the MATLAB® environment.

Besides the development of the previous detectors, the actual work will depict the scope of applications of the introduced periodicity features on more general species recognition systems. Additionally, an audio summarisation procedure is presented, condensing the bird-like sounds in a record, virtually without having prior knowledge on the recorded species' songs.

## 1.1 Related work

In an effort to concentrate the individual advances in the topic in focus, a first research network for computational bioacoustic monitoring and analysis has been established. The "International Expert Meeting on IT-Based Detection of Bioacoustical Patterns" was held at the Isle of Vilm in 2007 in Germany (see [FBC08] for the proceedings), and a bioacoustic monitoring mailing list has been set up [BML].

Although the previous discipline is rather young, most of the problems faced within the recognition of bird sounds constitute typical pattern recognition problems. In fact, the detection of a particular bird sound can be interpreted as the result of a classification procedure. A distinct classification result is associated with the recognition of a bird's voice, and the bird is detected. Hence, several common pattern recognition methods have been applied to the task of birdsong identification.

In particular, algorithms for speech recognition have proposed to be applicable on birdsongs: Anderson et al. [ADM96] have proven the Dynamic Time Warping approach to be suitable for the template-based detection of the songs of Indigo Bunting (*Passerina cyanea*) and the Zebra Finch (*Taeniopygia guttata*) in clean recordings. Furthermore, Kogan et al. have compared the previous results to the performance of hidden Markov models [KM98]. Here, a compound of left-to-right HMM's was used to analyse the recordings.

Both of the further procedures relay on manually extracted and segmented templates. In his Masters thesis [Fag04], Fagerlund, member of the Finnish AveSound project, describes a method to automatically derive a segmentation of birdsongs. Furthermore, a k-Means classificator is used for the final recognition task. As most of the previously mentioned studies were performed on spectral features, commonly used for the parametric representation of harmonic acoustic signals, Selin, Turunen and Tanttu used a Wavelet Packet Decomposition to extract a set of complementary parameters, permitting an improved description of inharmonic and transient bird sounds [STT07].

Focussing on the highly repetitive sound of crickets, Schwenker et al. [SDK$^+$03] used learnable Neuronal Networks to adapt and classify the insects' sounds, which have a great similarity to the songs of some birds of the *Locustella* family. Wavelets and Neuronal networks were also used for bird voice recognition in [Pos02].

A different point of view, considering the commonly used spectrogram in its absolute-value representation as an image, is applied in the work of Brandes et al.: before extracting a set of common spectral features, the spectrogram is processed using advanced image processing techniques. Here, the long element trains of frog and cricket calls are detected [BNF06].

## 1.2 Thesis overview

Basically, the topics discussed in this thesis can be subsumed in three major parts, focussing on the bioacoustic facts, their technical measurements and the evaluation of the latter in realistic scenarios.

The work on hand is focused on the analysis of signals being recorded in open biological environments. As a consequence of the chosen monitoring scenario, the derived recordings feature an heterogeneous set of acoustic events. Chapter 2 discusses the general censusing goals and strategies usually encountered in such scenarios whilst concentrating on the methods actually implemented in the thesis. Facilitating the description of birdsongs, the spectrogram is introduced as a basic tool for the representation of acoustic signals. Moreover, the avian headliners of this thesis are introduced to the reader. Here, the bioacoustic peculiarities of the envisaged species are in the center of interest.

In the following Chapters 3, 4 and 5, both the technical representation and recognition of birdsongs are discussed. In order to define the modes of representation used in the further parts of this thesis, a basic mathematical description of the applied signal processing techniques is given in Chapter 3. Here, the Fourier transform will be introduced, permitting a mathematical definition of the spectrogram. Now, several representations of acoustic signals, also called features, are developed in Chapter 4. As these representations are discussed in an increasing order of abstraction levels, the final goal of detecting a birdsong is continuously approached (see Figure 1.1):

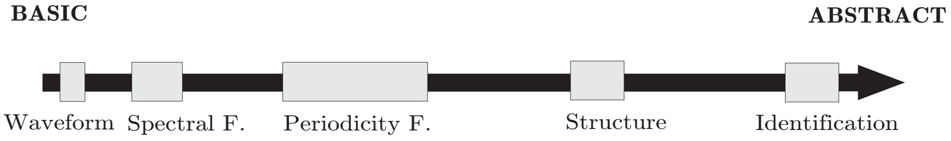


Figure 1.1: Representation of bird voices: relation of abstraction levels.

A set of spectral features is extracted directly from the spectral representation of an audio signal. Here, measurements commonly used in music information retrieval applications are adapted to fit the actual topic. Later on, these features will be used for the preselection of certain acoustic events. Furthermore, a more rough and thus quite robust spectrogram is derived. Focusing on birdsongs containing periodic structures, a set of more robust periodicity features is developed. Here, the spectrogram is analysed on the repetition of certain patterns. The robust nature of the periodicity features will ascertain the robustness of the deducted birdsong detectors.

Chapter 5 introduces the features developed above to their applications in bioacoustic pattern recognition systems. Following a methodological order, the distinct stages of the Chaffinch and Savi's Warbler detector will be described. Here, some techniques commonly used in multimedia retrieval applications, as Dynamic Time Warping, will be applied. Furthermore, some of the techniques developed for the detectors will be adapted to solve more general detection and analysis problems. Hence, an algorithm providing an automatic acoustic scene analysis is proposed. Moreover, several structure extraction methods, analysing the systematic character of birdsongs, are described.

In the last part of this thesis, the Chaffinch and Savi's Warbler detectors' performance will be evaluated. As referred in Chapter 6, a voluminous set of monitoring recordings was used to test the latter of the algorithms. The testing data was compiled from a large monitoring database, and special attention was given to the distribution of several noise conditions. Finally, in Chapter 7, the techniques being previously described are summarised. Considering the basic nature of some of these approaches, several improvements are proposed for future work. The thesis closes with some ideas on additional application fields of the periodicity features.

# Chapter 2

# Acoustic monitoring of bird activities

## 2.1 Unsupervised monitoring

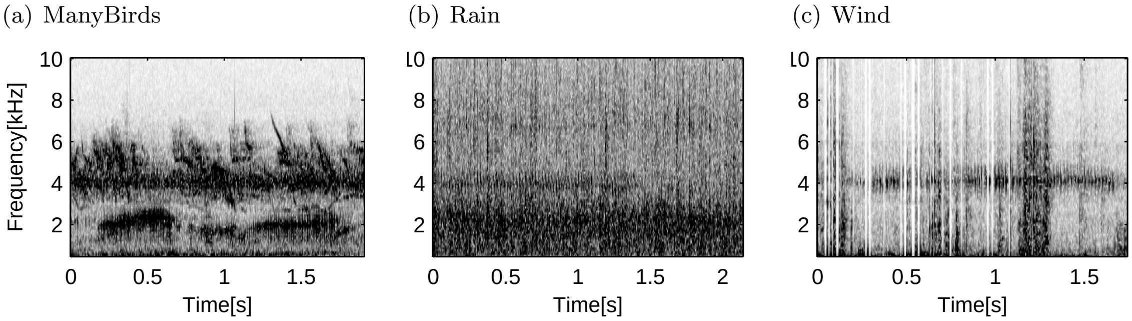
Bioacoustic monitoring, as described by Rempel et al. [RHH$^+$05], constitutes a useful extension to the methods used for the evaluation of songbird populations. Here, instead of ad-hoc accessing the desired population sizes afield, the statistics are extracted from long-time acoustic recordings. In a typical unsupervised monitoring scenario, a set of microphones is positioned in certain areas of interest. Fortunately, this intrusion can be scheduled preliminary to the typical breeding season of the birds. Thus, assuming the recording apparatus itself to be rather small, the interference on the recorded species' natural behaviour may be significantly reduced. In contrary to the usual ornithological practice, the setup of the recording apparatus can be realised by non-birders. Once set up, the microphones may record a significant amount of acoustic data. Here, the amount of recorded material is bounded by the capacity of the final recording medium. Using modern hard disk drives, the ultimate recording time is rather large, e.g. 375 hours of 4-channel CD-quality recordings fit on a 500GB disc. Moreover, triggering the recording events by either time schedules or acoustic events, the former capacity may be utilized more efficiently. Besides the previous limitation, the power consumption of the whole recording equipment states a critical factor on the system's running period. In the monitoring scenario described in Chapter 6.1, solar power was utilised to suspend and recharge the system's power reservoir.

As the setup of the monitoring microphones is usually static, the directionality and orientation of these sensors determine the actual acoustic field to be monitored. Here, the application of sensor-arrays promises a more flexible approach: subsequent to the recording, the accumulated audio signals may be focussed on arbitrary positions using acoustic beamforming techniques, as described in [VB88]. Most of the recordings considered in this thesis were picked up using 4-channel arrays of cardioid microphones.

Unlike the usual recordings performed by ornithologists, often being focussed on a single individual of interest, monitoring recordings usually contain an assemblage of acoustic events (see Figure 2.1a). In many cases, the volume of an envisaged species is outperformed by another animal. Also, besides the sounds made by neighbouring animals, a great influence is given by the noises produced by our technology. This is especially the case when recording in urban environments. Here, a complex mixture of diverse noise sources complicates the automatic detection of bird voices. Given a more natural environment, the main sources of noise are represented by wind and rain (see Figure 2.1b,c). Moreover, the noise of planes and trains contributes to the overall noise floor.

As a further factor, the influence of alterations of the acoustic signal, being induced by the environment, is a serious issue when recording in a monitoring setting. Especially in forests, the echo and reverb effects produced by the trees, reflecting the animal sounds, have to be considered when developing an automatic recognition system. In urban areas, these effects are also prominent in places being surrounded by concrete walls. An additional aggravation is given by the animals' distances to the microphones being almost unrestricted. Thus, the echo of a birdsong may easily arrive at the sensor with a volume being comparable to that of the direct signal. Furthermore, given the typical low-pass character of air, birdsongs recorded from greater distance lack some of their typical harmonic components. Thus, the amount of information being usable for an automatic detector is reduced.

As shown in the previous paragraphs, although the material being derived during a monitoring session may be reused for various applications, the extraction of clean parameters for individual acoustic events states a challenge. Thus, the robustness of the envisaged detectors will be a main factor affecting the performance of the detection algorithms. In this thesis, the robustness of the detectors will be based on robust feature extraction mechanisms. In other words, the goal of the proposed routines is to extract parameters being characteristic to the targeted birds' songs, whilst keeping the extracted values invariant up to the distortions mentioned above.


(a) ManyBirds

(b) Rain

(c) Wind

Figure 2.1: Spectrograms of typical monitoring recordings for three typical environmental conditions.

## 2.2 Detection goals

As the algorithms to be developed in his work have their main applications in the areas of biological censusing and biodiversity assessment, the output of the procedures should suit the specific requirements. Actually, the mentioned topics require different modes of automatic acoustical scene analysis. In order to develop a suitable automatic detector, the following demands have to be evaluated:

- Definition of the elementary *detection subject(s)*: detect single individuals, certain groups, or build a generic detector for a species.

- Specification of the *detection accuracy*: detect all/some songs of a bird. Allow some/no false detections.
- *Quantification* of the *detection classes*: simultaneously distinguish between multiple species or individuals?

From the computational point of view, the mentioned detectors are represented by classification routines. Here, a simple detector, designed to recognise a distinct species, is realised using a binary classifier. For example, in the Chaffinch detector, the monitoring recordings are partitioned in segments classified as either “non-Chaffinch” or “Chaffinch”.

This concept is also applied on the Savi’s Warbler detector, but, in contrast to the Chaffinch detector, two operational modes are implemented for the latter algorithm. When set to the first mode, the detector is only applied to some particular segments determined in a preprocessing step. Thus, some computation time is saved, but the detector is likely not to find all occurrences of this bird’s song. Still, if the actual focus is set on a raw estimate of the Savi’s Warbler’s presence, the output of this algorithm should prove to be sufficient. On the other hand, a second operating mode provides the means for annotating almost all occurrences of Savi’s Warbler songs.

For the detectors mentioned above, the internal signal representations, including the features to be described in Chapter 4, are designed to reflect the parameters being most descriptive for the envisaged species. Moreover, the classification criteria are fixed on the observation of distinct parameter ranges. Thus, the detectors are only suitable for scenarios where a single species has to be detected. In order to perform the detection of multiple species, the feature set has to be reconsidered. Here, the features have to reflect some parameters of each of the envisaged species. Furthermore, the new feature design must allow a discrimination of these species. In Chapter 5.3.2, a feature set designed for such a multi-species detector is introduced. The actual classification is usually performed by a learnable classification system as a Neuronal Network or a hidden Markov model. This allows for an automatic evaluation of classification parameters, which is, however, beyond the scope of this thesis. Several remarks are given at the appropriate positions, proposing further applications including such classification systems.

Considering the accuracy of the Savi’s Warbler and Chaffinch detectors, the algorithms were designed to achieve a minimum percentage of false positives. Thus, the number of segments being falsely classified as Chaffinch songs is minimised. Unfortunately, as more candidates have to be discarded, this usually leads to an increased amount of false negatives. As described in 6.2, this is also the case for the Chaffinch detector. Then again, as mentioned at the beginning of this section, the requirements of a different censusing approach may enforce the minimisation of the percentage of false negatives. For example, if all performed stanza’s of a Chaffinch shall be manually analysed, the increased amount of false positives can be easily reduced by the user performing the manual postprocessing. Thus, the algorithm may be parametrised to return a more vaguely classified set of song candidates.

## 2.3 The acoustic nature and representation of birdsongs

Being part of the natural acoustic background since the beginning of the human race, birdsongs are often conceived as little more valuable. Thus, today, the usual perception of a common birdsong may be compared to the superficial perception of a warehouse radio. Anyhow, the songs of some common birds are easily identified by many people. Physically represented as an ordinary acoustic wave, the song of a bird can only be recognised as such by an active learning effort. Here, the problem of an "adequate" representation arises. Actually, the usability of a distinct representation is mainly determined by the intended recipient of the chosen representation. Thus, a bird's internal representation of its song may be completely different from the way we perceive it. Focusing on a representation suitable to the author and readers of this thesis, and thus neglecting the intentioned receiver of the discussed signals, the name of the topic itself motivates a first approach: bird - song.

Availing oneself of the opportunity of listening to a birdsong with the suitable interest, the reader may agree that both music and birdsongs allow the introduction of parameters like dynamics, tonal variance and timing. Moreover, the introduction of certain sequences of fixed entities, as found in a melody or rhythm, can easily be transported to the structured nature of many birdsongs. Thus, the terms of a bird's melody, the Woodpecker's rhythm or the Chaffinch's song are based on intuitive relations. The long tradition of this concept is underlined by examples as of Athanasius Kircher noting down the voices of several birds by means of short musical scores in 1650. Here, a Chaffinch song, mimicked in Olivier Messiaen's "Catalogue d'oiseaux" will be used as an example to depict the potential of such a representation.

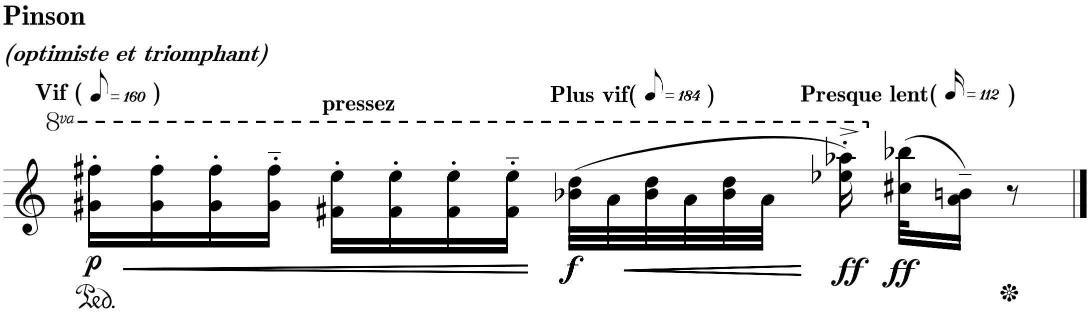


Figure 2.2: Excerpt from Messiaen's "Catalogue d'oiseaux, XI", containing a Chaffinch motif. The left-hand part of the score is omitted.

Composing the piece in the middle of the last century (1958), whilst being versant with the structural tools found in contemporary serialism, Messiaen uses a wide range of notational and textual symbols for describing this common bird's song. Here, instead of tying the song to a simple, proven structure, the birdsong's structure is attempted to be left untouched, establishing its inherent structure instead.

Considering the scientific ornithologists, the musician, writing the bird's score, was substituted by an apparatus some years before. The spectrogram, with the sonograph as its predecessor, notating the bird's voice by means of a time-frequency diagram, replaced the musical score by referring loudness, tonal frequencies and rhythms in continuously variable parameters. Thus, the limitations of the historic musical notation system were overcome by a more physical low-level representation of the birdsongs.

In Figure 2.3, a spectrogram of a typical Chaffinch's stanza is depicted. The colour of each dot in this image represents the energy contained in a distinct frequency band at a distinct time interval. As indicated by the reference scale shown in Figure 2.5, black dots represent

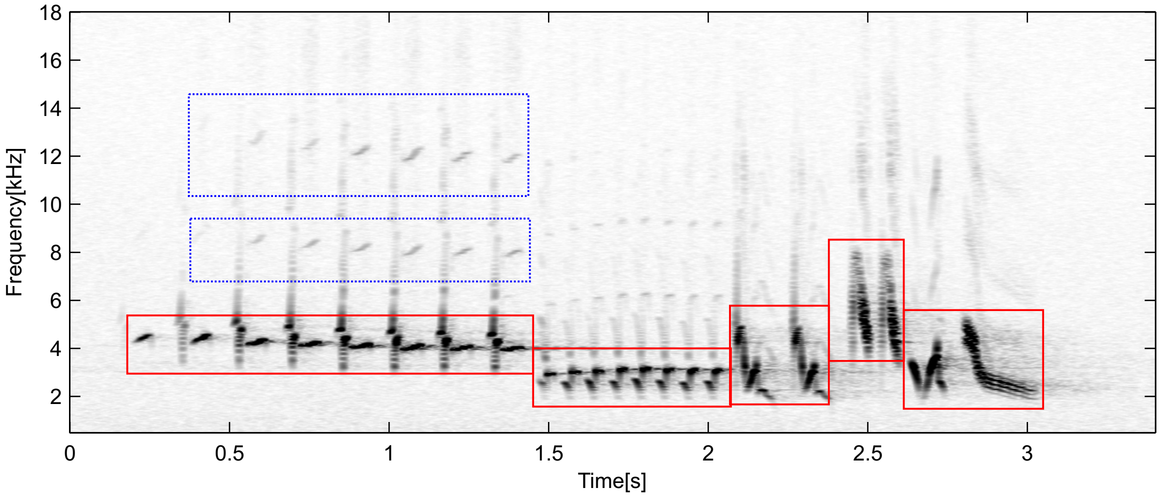


Figure 2.3: Spectrogram of a Chaffinch's (*Fringilla coelebs*) stanza.

high energy measurements for the respective coordinates. The main structure of the song might thus be sketched by drawing the sequence of dark black lines in a range from 2-6kHz containing the majority of the depicted signal's energy. These lines carry what is called the "main melody" of the song, as highlighted by the lower red boxes. Considering the continuous frequency progression within these single lines, especially at the end of the stanza, the spectrogram can still be taken as an instruction for whistling the Chaffinch song. Moreover, the above spectrogram features a simultaneous sketch of the melody in the higher frequencies (blue, dashed boxes). As it is the case for the Chaffinch's song, these components are caused by harmonics: multiples of the frequencies contained in the main melody, being generated by the acoustic characteristics of the bird's singing apparatus.

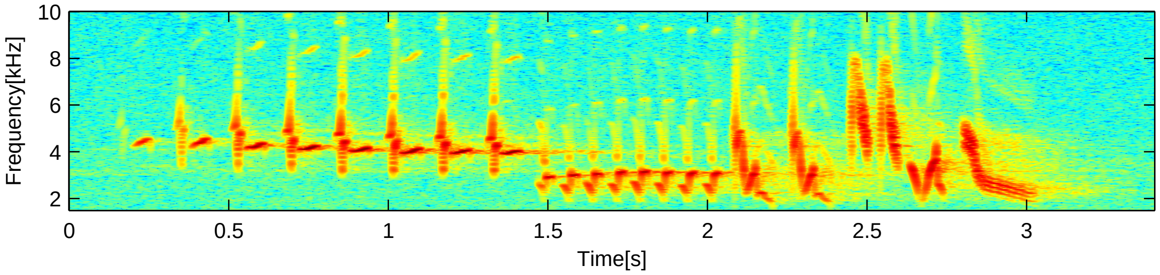


Figure 2.4: Colored spectrogram of a Chaffinch stanza. Frequency band: 1.5-10 kHz.

The above Figure 2.4 images a coloured excerpt of the previous example. The black-white spectrograms, focussing on high-energy components of a signal, are more easily interpreted by the reader. Thus, in this work, non-coloured spectrograms will be used where applicable. However, when the simultaneous representation of both foreground and quiet background signals is demanded, the jet colour map is used for the representation of signal energies. Using this colour space, being larger than the black-white parametrisation, when representing spectrograms and features, noisy signals are depicted more accurately. The relation of signal energy and image colours is depicted in Figure 2.5.

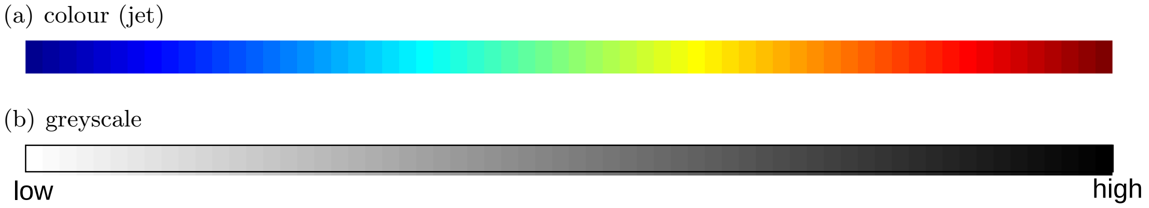


Figure 2.5: Colormaps used for spectrograms and feature representation.

In this work, the spectrogram, being widely used for the description of acoustic signals, will represent the different acoustic examples to the reader. With little training, the identification of some bird sounds as well as the imagination of their acoustic impressions can be accomplished by many people. In contrast, the digital representation of a spectrogram is very difficult to interpret for the machine. Thus, representing the main topic of this work, complementary representations have to be introduced, measuring some basic as well as advanced parameters of the analysed signal. These features, introduced in Chapter 4, and the outcomes of the following analysis steps are explained in an order of ascending abstraction levels, as depicted in Figure 1.1.

## 2.4 Chaffinch (*Fringilla coelebs*, Buchfink)

In Europe, the Chaffinch or *Fringilla coelebs* is a widespread and very familiar songbird. The IUCN Red List of Threatened Species estimated the number of individuals populating Europe to 270-480 million birds [IUC] in 2004. Its song, although being quite complex, is easily identified by many people. This, on the one hand, may be caused by the quite frequent appearance of the bird and its song, which, considering the actual situation in Germany, can be heard at least since March. On the other hand, examining the spectrograms of different Chaffinch stanzas in Figure 2.6, the idea of an underlying structural similarity may arise. At first it has to be noted that the excerpts shown in the respective figure are drawn from different recordings of different individuals. As the reader may have noticed by listening to the song of the about 15 cm tall bird, the stanza instantiations of a single individual, being sung in a close time range, are very similar to each other. In addition to the mentioned structural form, this precise similarity, in the very first encounters to this birdsong, made the author reckon to work on an relatively manageable amount of variations. But, far from it, after analysing the recordings at the Animal Sound Archive at the Berlin University, it was clear that approaching the development of a generic detector for this species required a quite general description of a Chaffinch song's structure.

In order to achieve some knowledge about common song parameters, a survey, being discussed in detail in the next section (2.4.1), was carried out. Thereby, a special focus was set to the usual segmentation of the individual stanzas. This segmentation is motivated by the usually repetitive structure of a Chaffinch's stanza, constituting groups of repeated spectral entities. This structure is already depicted in Messiaen's musical representation of a Chaffinch stanza, depicted in Figure 2.2. An almost canonical finishing flourish, as performed by all of the six individuals analysed in the respective figures, was identified, surely facilitating the recognition of the bird's song by humans. Furthermore, the overall frequency range used by the respective birds was found to be bounded by about 2 and 8 kHz, respectively, although some harmonics of the tones being sung easily exhaust the frequency range audible to humans. Beside these common factors, the songs of two individuals may differ by many factors at different structural levels.

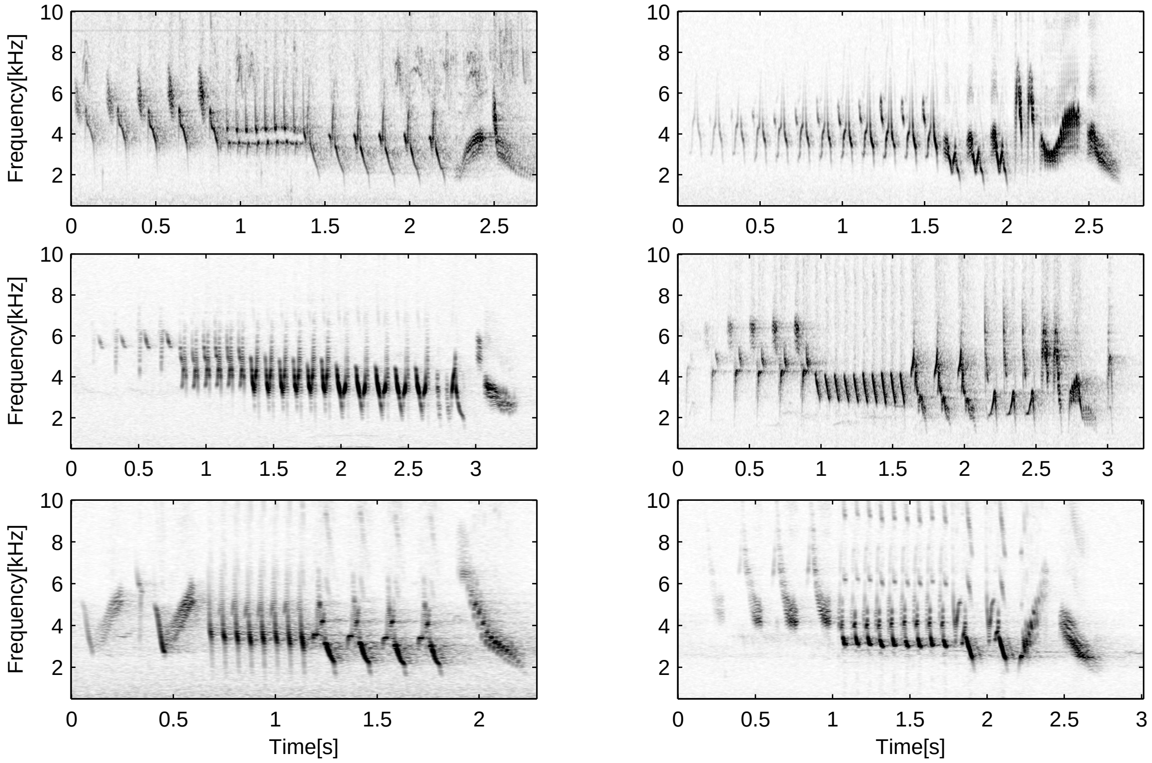


Figure 2.6: Spectrograms displaying song stanzas of 6 different Chaffinch individuals.

### 2.4.1 Song structure

In order to describe the song of the Chaffinch in a detailed way, a hierarchical terminology describing the components of a song will be introduced first. Regarding the existing literature, the meaning of the proposed terms seems slightly variable. In this work, the definition of the following components is based on those used by Bergmann and Helb [BH82]. As the goal of this chapter is to describe the structure of the Chaffinch's song, we will take this song as an example for a typically structured birdsong. Actually, being well structured, the Chaffinch's song is often used for this purpose.

It is obvious, that timing and temporal structure are of great importance for the singing behaviour of birds. On a large scale view, a temporal constraint is given by the seasons in which the Chaffinch usually sings. As with most of the other songbirds, these periods are centred around the breading season of the particular animal. Within a singing season, the Chaffinch concentrates the performance of its *songs* on the times of dawn and sunset. Each dawn, several songs are sung by the bird. In the case of the Chaffinch, these songs are instantiated through the repetitive vocalisation of song *stanzas*, which are recognized as the bird's melody or theme. Each of these stanzas takes as long as about 2-3 seconds, and a space of about 6-12 seconds is usually left between two consecutive stanzas.

As many songbirds are able to adapt their singing behaviour to short- and long-time disturbances of their habitat, for example placing their stanzas in moments of silence as well as by singing at night when the city noise floor is quite low, the quantifications given in this chapter are meant to be estimates. Despite these deviations, the timing of the repetition noted above is astonishingly accurate.

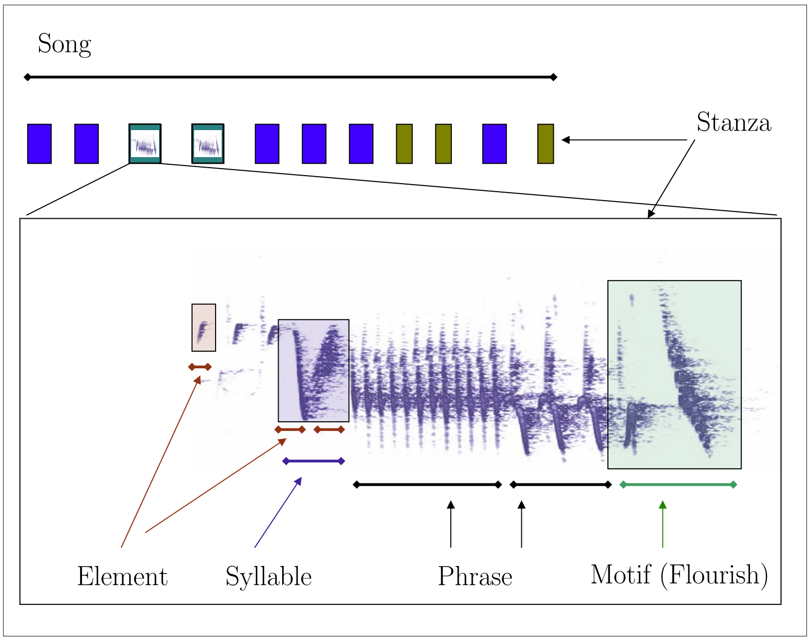


Figure 2.7: Hierarchical structure of a Chaffinch's song.

The definition of the song's fine structure, as depicted in Figure 2.7, is usually performed in the spectral domain, using the spectrogram together with an acoustic impression to create symbolic definitions. Each *stanza* is defined to be a successional construct of phrases, syllables and elements. Here, the *element* as the smallest, atomic unit, resembles a short and coherent line or blob in the spectrum. Common parameters used to describe an element include amplitude, center frequency, time and frequency dimensions as well as a direction, indicating a gradual increase or decrease in frequency. A *syllable* is usually built from two different elements sounding in very close succession. Actually, the listener may perceive a syllable as a single element. If an element or syllable is quickly repeated several times without interruption, the repeated instantiations form a *phrase*. Here, the repetition frequency or the period in which the elements are repeated, will built an elementary feature to be used in the algorithms proposed in this thesis. In Figure 2.8, this frequency is annotated for the phrases of a Chaffinch's song. Observing the song of a Great Tit or Chiffchaff, a last structural element is introduced. The *motif* is defined to be constructed from two or more elements, which, to a listener, appear to be grouped but clearly distinct. In the special case of the Chaffinch's stanza, many instantiations end with a characteristic motif, also described as *flourish*. In Chapter 5.2.2, this flourish will be used in a candidate extraction routine for the Chaffinch detector. As the above definitions are subject to the listener's perception, a certain amount of ambiguity is to be expected.

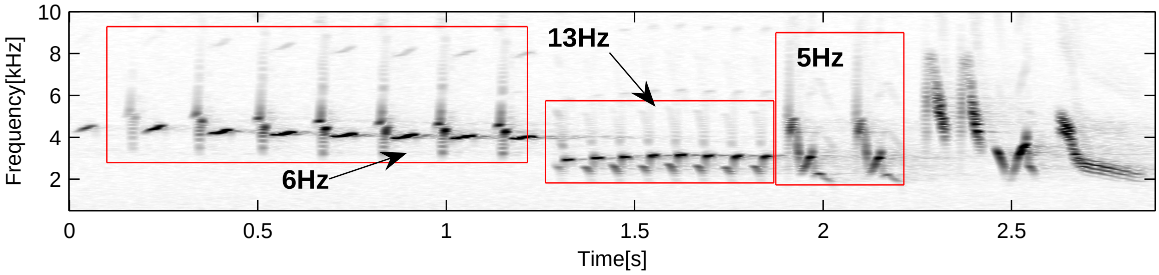


Figure 2.8: Chaffinch stanza with highlighted phrases and annotated element repetition frequencies. The final, unboxed repetition is interpreted as part of a flourish motif.

### 2.4.2 Stanza segmentation study

In order to derive an abstract model of the Chaffinch's stanza, to be used for designing the classification routine described in Chapter 5.4.1, a small study was carried out in order to get statistical data on the typical structure of the songs of this common bird. In this study, 115 stanzas from over 50 Chaffinch individuals, obtained from the Animal Sound Archive of the Humboldt University, Berlin, were manually analysed by inspecting the extracted spectrograms. The 115 stanzas were selected from a set of over 800 available stanzas. These being grouped according to the recording they were contained in, the stanzas used in this survey were picked randomly, but with a forced limit of a maximum of 3 stanzas per recording.

At first, each stanza was divided into a series of segments. A distinction between three types of segments was made in this process: The segment type of main interest (type-I) is the "phrase" type, corresponding to a single phrase as defined above. This type of segment is defined to be delimited by significant changes either in the shape of the repeated syllables or of the actual element repetition period. The latter case is exemplified in Figure 2.9, considering the 3rd and 4th segment.

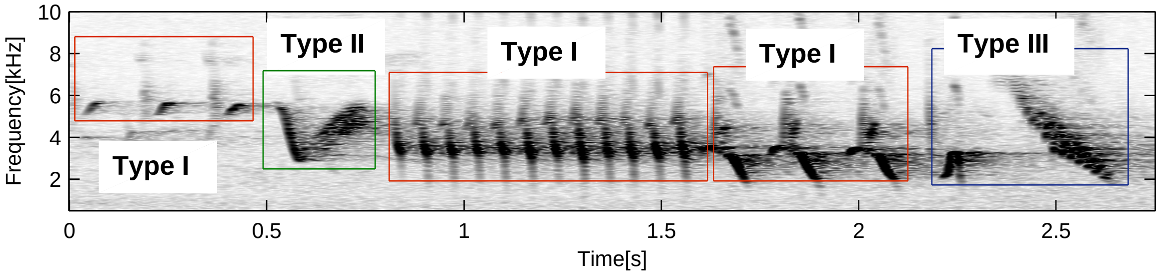


Figure 2.9: Chaffinch stanza with highlighted segments. The segment types used for the study are also annotated. The first segment, as well as with the two in the middle, are counted as type-I. A single motif is annotated as type-II. The flourish typical for this species is represented by the type-III category.

The remaining two types were both used to identify extracts fitting the definition of motifs. Thereby, the type-III segments were associated to typical Chaffinch flourishs at the stanza-tails. Other non-periodic segments were subsumed as segments of type-II. Now, for each stanza, the total number of segments, and the following individual segment parameters were measured:

- Segment type,
- Temporal position within the stanza,
- Segment length,
- Relative loudness regarding the whole stanza,
- Element number and repetition period of type-I segments.

In the following paragraphs, a selection of those results being relevant for the elaborated algorithms are presented. Performing the manual segmentation without leaving any space between the segments, the length of the whole stanza can be reconstructed from the individual lengths of its segments. For the calls analysed in this study, the mean stanza length amounts to **2533** ms with a standard deviation of **440** ms (17%) and extreme values of **1460** ms and **3720** ms, respectively. Considering the segment volume, the variation was found to be negligible, though 5 of the first call segments, mostly being high pitched, were recorded less loud than the rest of the related segments. This quiet stanza beginning is also represented in Messiaen's Chaffinch score (Fig. 2.2).

Considering the number of segments usually found in a stanza, the following Figure (2.10) reveals that the majority of stanzas contain 4-5 segments:

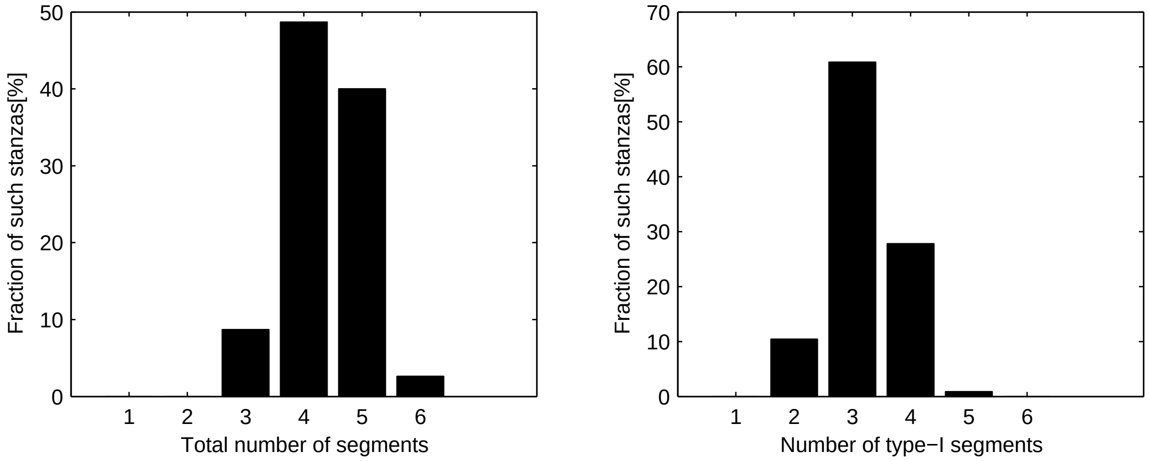


Figure 2.10: Two statistics on segment distribution. Left: percentage of segments extracted from stanzas. Right: amount of stanzas having a specific number of type-I segments in %.

Only counting the type-I segments, it is obvious that the majority of all segments belongs to the phrase category. Usually, one or two non-periodic segments can be found in a stanza. As it was ensured that all evaluated stanzas contained the typical type-III finishing flourish, we can conclude that **73%** of the analysed calls had no further non-periodic segment of type II. In the classification routine to be developed in section 5.4.1, only a single such segment is allowed for a successful detection. Thus a maximum of **89%** of this survey's test set is covered by the referred procedure.

Considering the periodic segments, an average number of 5 elements is repeated in each stanza. The mean repetition frequency was measured to **8.8**Hz. Both of the previous values are measured with standard deviations around **55%**. This rather big variance is established by the extreme values for the element repetition frequency being measured as **2.6** Hz and **36** Hz, respectively. A more detailed analysis was performed on the subset comprising the stanzas that feature a total of 4 segments, whereas 3 of these are classified as type-I. Such stanzas are printed in the two bottom-most spectrograms shown in Figure 2.6. As the last segment is always of type-III, the mentioned stanza class is described by the succession of the segment types I-I-I-III. Representing **46%** of the whole test set, the element repetition frequencies of the individual segment positions were analysed for this subset.

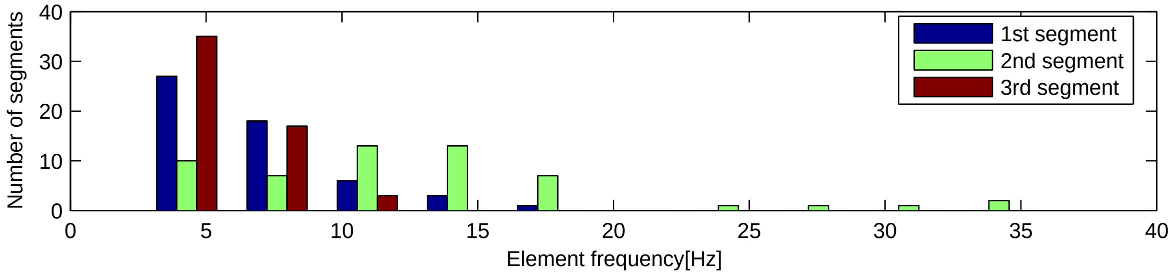


Figure 2.11: Histogram showing the distribution of element repetition frequencies on the segment positions. Only type I-I-I-III stanzas were analysed.

In the period histogram (Fig. 2.11), the three relevant segment positions' typical element repetition frequencies are analysed. As the final flourish is assumed to be aperiodic, the corresponding position is ignored in this step. Most of the element repetition frequencies are located around 7-8 Hz, although the second segment, being located in the midst of the call, features lots of element repetition frequencies being significantly greater. The other two segments' frequencies are bounded by about 15 Hz (see Table 2.1). This phenomenon, which may be partially based on a "slow start" of the singing bird, will lead to a minimum requirement on the inter-segment element frequency alternation used for our Chaffinch detector, see Chapter 5.4.1.

| **Seg. position** | **1** | **2** | **3** |
|---|---|---|---|
| **Mean** | 6.6 | 13.1 | 6.1 |
| **Std** | 3.1 | 7.0 | 1.5 |
| **Min** | 2.6 | 4.6 | 3.3 |
| **Max** | 16.3 | 35.9 | 11.2 |

Table 2.1: Element repetition frequencies for individual segment positions. Mean, standard deviation and extremal values in Hz. Only 4-segment stanzas were analysed.

Finally, the individual segment lengths were analysed. Similar to the results for the repetition frequency, the mean segment length of **580** ms is drawn from a strongly inhomogeneous set, featuring a standard deviation of **60%**. Thus, the smaller (type I-I-I-III) subset, used for the periodicity analysis above, was also used for a detailed analysis of the position-separated segment lengths. In the corresponding histogram (Fig. 2.12), the long, continous-frequency segments, which are likely to occur at the beginning of a stanza, are pushing the segment length for this first position beyond the usual length measurements. A call featuring such a segment is displayed in the top-right corner of Figure 2.6.

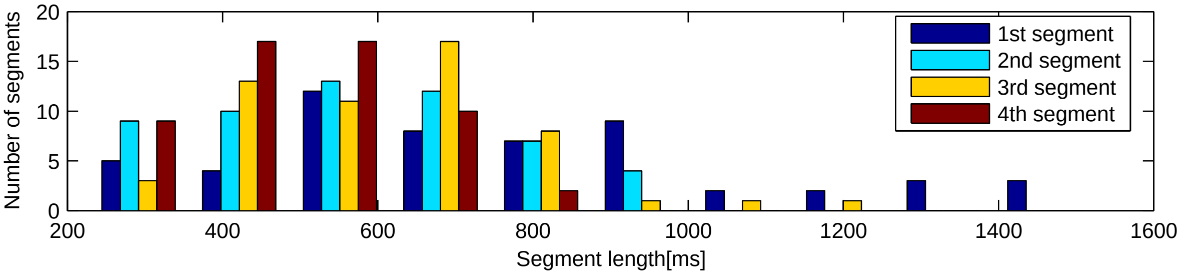


Figure 2.12: Histogram showing the distribution of segment length on the segment positions. Only type I-I-I-III stanzas were analysed.

| Seg. position | 4 | 3 | 2 | 1 |
|---|---|---|---|---|
| Mean | 495 | 616 | 579 | 770 |
| Std | 134 | 188 | 195 | 315 |

Table 2.2: Segment length statistics for individual segment positions. Mean, standard deviation and extreme values in ms. Only 4-segment stanzas were analysed.

## 2.5 Savi's Warbler (*Locustella luscinioides*, Rohrschwirl)

Being discovered by the Italian geologist and ornithologist Paolo Savi, the subsequently addressed singing bird is called Savi's Warbler. In contrast to the Chaffinch, Savi's Warbler, inhabited in Europe, has a small population of about a million individuals [IUC]. Living in wetland habitats, this bird uses the reed bed to both sing and collect its food. Due to its colour, which is well adapted to the typical texture of its habitat, and the small size of about 14 cm, the bird is usually well hidden. In order to collect data on this bird's population, visual identification turns out to be quite difficult given the short distance necessary to identify the bird. However, the song of this small bird figures out to be very characteristic, although it may be confused with the call of Roesel's Bush-cricket. The Warbler's cricket-like song, enduring from call-like 3 seconds to a timespan of several minutes, is based on the fast repetition of two similar elements, as shown in the left column of Figure 2.13. Here, fast repetition means an element repetition frequency between 45 and 55 Hz. Considering the spectrogram, Savi's Warbler's song is represented by a steady cloudy stream, peaking at about 4.2 kHz, which contains the mentioned element train. Although this click-train features several higher frequencies with peaks around 7 and 14 kHz, these components are quickly lost when recording a more distant bird. In Figure 2.13, the right column shows a two channel excerpt taken from a four-channel monitoring recording. Here, two warblers are taking turns in singing. Although the song of the first bird, being best recorded in the first channel (upper figure), features frequencies exceeding 6 kHz, the second channel (lower figure), recording the second bird from greater distance, does not show such high frequency components. Note although the center frequencies of both individuals differ by 200 Hz, their element repetition rate is almost identical.

Considering its general singing behaviour during the breeding season, the Warbler is easily captured on tape: when domiciled in the recording area, the single individuals are likely to be recorded during their several hours of active singing per day. In general, there are two types of songs being sung by the bird. The typical song can last more than 20 minutes, featuring only

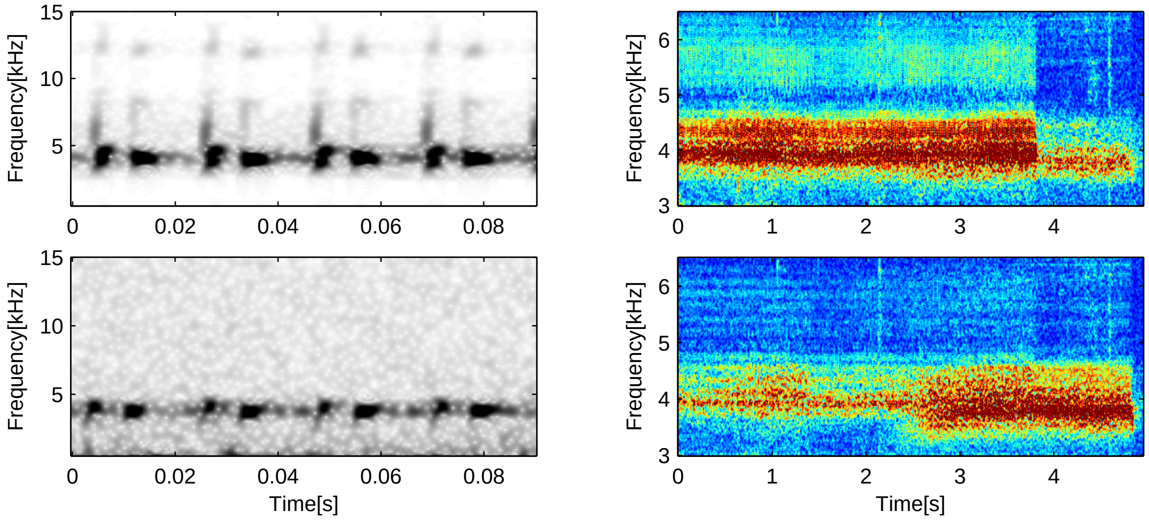


Figure 2.13: Left column: Two spectrograms, each displaying the elements of a Savi's Warbler's song, with high time resolution. The bottom figure shows a nightly song, recorded at great distance. In order to depict the gentle sound, the latter signal has been amplified. Right column: Spectrograms of a stereo recording featuring two Warblers' songs.

short breaks. Listening to the recordings made at Lake Parstein, Germany, the simultaneous singing of two or even more Savi's Warblers is a common event. Besides these long songs, some short phrases are often added to the simultaneous song of another Warbler. As both of the mentioned song types feature the constant 50 Hz element frequency, the Warbler has been called a "flying oscillator", the detector introduced in Chapter 5.3.1 will be based on a robust detection of this oscillation.

# Chapter 3

# Audio signal processing background

Having introduced the target signals to be detected as well as the conditions under which this detection has to be performed, we now switch to the computational point of view. In this chapter, some basic mathematical requirements and techniques will be introduced, as a basis for understanding the subsequent feature extraction routines.

At first, a mathematical representation of (audio) signals is introduced. This allows for a quick discussion of the audio digitalisation procedure and its effects on the sampled signal. Furthermore, the Fourier transform, yielding a complementary representation of the audio signal and revealing its properties in the frequency domain, is introduced. The convolution operator is introduced and, exploiting the properties of the above transform, the fast convolution method and its application to digital filtering will be derived. In this thesis, digital filters will be mainly used for the purpose of smoothing or low-pass filtering signals. Finally, the spectrogram, as introduced in the previous chapter, will be defined on the basis of the windowed Fourier transform.

## 3.1 Notation and mathematical symbols

In the following, the symbol ”$\cdot$” is usually used for scalar multiplication. Furthermore, when used as a parameter to a function or vector, it denotes the free parameter as in

$$x(\cdot) := (\dots, x(-1), x(0), x(1), \dots)^\top.$$

In between two vectors or functions $x, y : D \to \mathbb{R}$, for $D \subseteq \mathbb{Z}$ or $D \subseteq \mathbb{R}$ the symbol ”$\cdot$” denotes the pointwise multiplication of $x$ and $y$:

$$(x \cdot y)(k) := x(k)y(k),\ k \in \mathbb{Z}.$$

## 3.2 Audio signals

Physically, an audio signal is an interpretation of an acoustic pressure wave, propagating along various carrier substances with a certain speed. In bioacoustics, the most popular carriers are water and air, the latter also being the usual case with songbirds. Finally arriving at the diaphragm of the monitoring microphone, the song emitted from a bird will have undergone several acoustic modifications described in section 2.1. Now, the type of this microphone will finally affect the translation of air pressure modulation into an electric signal. Here, the angle of the sound that arrives at the sensor, as well as the microphone's directional characteristics and frequency response, will be of importance.

The previous instantiations of an acoustic signal can be mathematically expressed as functions $f : \mathbb{R} \to \mathbb{R}$. Observing the change of air pressure from a fixed point of view, the domain $\mathbb{R}$ represents the time axis, and the range $\mathbb{R}$ measures the relative deviation from normal air pressure. In the electric domain, the latter expresses the amplitude of voltage change transmitted by the microphone. The signal's *waveform* is thus represented by the graph of its function $f$.

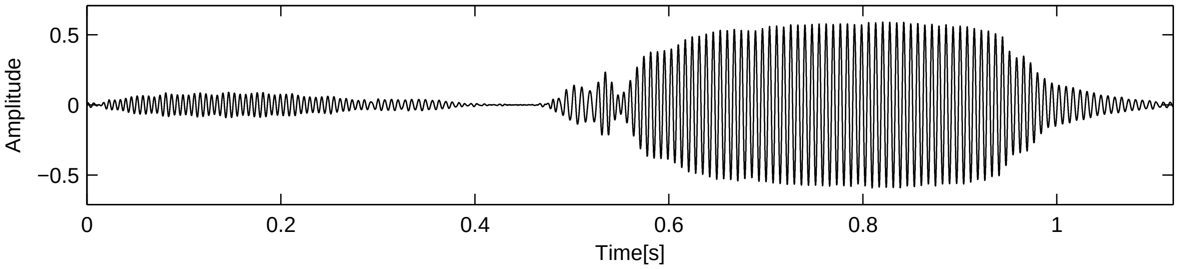


Figure 3.1: Voltage alternation induced by the call of an Eurasian Bittern (*Botauris stellaris*).

### 3.2.1 Sampling

As the electric sound signal reaches the audio card of a digital recording hardware system, it is converted into its digital representation by the A-D converter. Most commonly, audio signals are stored as pulse code modulation (PCM) files. These represent the signal as a series of sampled signal values, corresponding to precisely timed voltage measurements. The samples can be obtained using sampling and quantisation techniques, which realize the discretisation of an analogue signal in the time and energy domain. A sampled and hence discrete time signal is mathematically expressed as a function $x : \mathbb{Z} \to \mathbb{R}$, where $\mathbb{Z}$ depicts the equidistantly sampled time domain. Let $f$ be a continuous time signal. These sampled signal is then defined to be $n \mapsto f(T \cdot n)$, for $n \in \mathbb{Z}$. $T$ is called the *sampling rate.* Sampling a signal, it is of significant importance to choose an adequate sampling rate $T$ or sampling frequency $f_s = \frac{1}{T}$. According to the Nyquist theorem, a $T$-sampled signal can only be perfectly reconstructed if its frequency bandwidth is bounded by $\frac{1}{2 \cdot T}$. Hence, the signal is usually cleaned from components exceeding this threshold by means of low-pass filtering it in a preprocessing step. As the filtering mentioned above cannot perfectly erase the overlapping frequencies, more or less small errors, known as alias signals, are introduced during the sampling process. This is especially the case with time-limited signals. In Figure 3.2, the Bittern signal shown above is sampled using a sampling frequency of 500 Hz. As the call of this bird does not contain significant energies in frequencies above 170 Hz, the sampled approximation is quite precise. In order to guarantee an appropriate representation of the bird voices considered in this work, the utilized audio recordings are stored using a minimum sampling frequency of 48 kHz, consequently containing information of frequencies up to 24 kHz. Thus, during the recording, a voltage measurement is made every $41.\bar{6}$ microseconds. In the following, this sample rate will be used as a reference for all sample time related values.

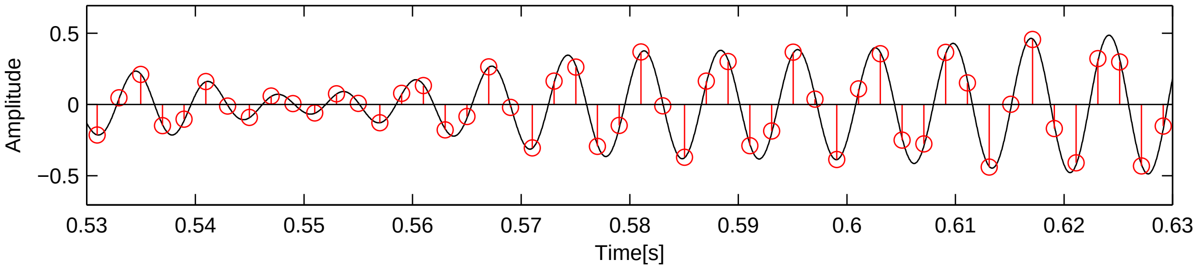


Figure 3.2: 100 ms excerpt from the Bittern signal of Fig.3.1, sampled at 500 Hz. The sampled values are indicated by red stems.

For the above definition of a discrete-time signal, the superposition of two sampled signals $x, y : \mathbb{Z} \to \mathbb{R}$ is defined by the pointwise sum $(x + y)(t) := x(t) + y(t)$. Amplification of an audio signal by a factor $\lambda \in \mathbb{R}$ is expressed through scalar multiplication: $(\lambda f)(t) = \lambda f(t)$. The resulting vector space $\mathbb{R}^{\mathbb{Z}}$ is very large and contains bad signals with infinite energy or sample values. Thus, one usually restricts the set of described signals to the Lebesgue spaces $\ell^p(\mathbb{Z})$, which are defined as follows:

Let $1 \leq p < \infty$ be a real number. The Lebesgue space $\ell^p(\mathbb{Z})$ is defined as

$$\ell^p(\mathbb{Z}) := \left\{ x : \mathbb{Z} \to \mathbb{C} \,\middle|\, \sum_{t \in \mathbb{Z}} |x(t)|^p < \infty \right\}. \tag{3.1}$$

The Lebesgue space $\ell^\infty(\mathbb{Z})$ contains all signals with bounded sample values:

$$\ell^\infty(\mathbb{Z}) := \{x : \mathbb{Z} \to \mathbb{C} \mid \exists B > 0 : \forall n \in \mathbb{Z} : |x(n)| \leq B\}. \tag{3.2}$$

The maps

$$\begin{aligned} \|x\|_p &:= \left( \sum_{n \in \mathbb{Z}} |x(n)|^p \right)^{1/p} \quad \text{for } 1 \leq p < \infty, \\ \|x\|_\infty &:= \sup\{|x(n)| \mid n \in \mathbb{Z}\}, \quad \text{and} \\ \langle x, y \rangle &:= \sum_{n \in \mathbb{Z}} x(n)\overline{y(n)}. \end{aligned} \tag{3.3}$$

define a norm and the scalar product on $\ell^p(\mathbb{Z})$ and $\ell^\infty(\mathbb{Z})$, respectively. These spaces are complete with respect to the norms and therefore constitute Banach spaces.

Allowing the signal $x$ to be a complex valued function is very useful for describing some important mathematical findings. A real-valued signal $x : \mathbb{Z} \to \mathbb{R}$ is easily extended to $\mathbb{C}$ by defining the imaginary part to be zero. For the above $\ell^p$ spaces, intuitively, the parameter $p$ controls the roughness of the contained signals. The smaller the value for $p$, the less samples with large amplitudes are allowed. Moreover, the $\ell^p(\mathbb{Z})$ spaces are nested: for $1 \leq p < q \leq \infty$, $\ell^p(\mathbb{Z}) \subseteq \ell^q(\mathbb{Z})$ holds. Note, that for $p = 2$, the Lebesgue space contains only signals with finite overall energy. In general, audio recordings of finite length and thus having a finite domain $I \subset \mathbb{Z}$, are extended to $\mathbb{Z}$ by assuming the unknown parts of the signal to be zero. Hence, these signals, being bounded sample wise, are contained in the space $\ell^2(\mathbb{Z})$, which now will be in the centre of our interest.

### 3.2.2 Quantisation

As the limited memory of a machine is incapable of representing the whole amplitude range $\mathbb{R}$, each sampled value is approximated by an integer of fixed precision. In general, a uniform quantiser is used to quantise the amplitude values at the sampled time positions. To this end, a maximum measurement value is defined, and the residual range is divided into uniformly spaced intervals. Each sampled energy value is represented by a 16 or 24 bit integer, corresponding to the interval containing the measured value. Note that this discretisation step is generally lossy for real-valued analogue signals. In contrast to the sampling process, it is not possible to reconstruct a signal from it's quantized version. An example displaying the extremely rough quantisation of the Bittern sample is provided in Figure 3.3. For an ideal uniform quantiser, the quantisation noise is estimated to a minimum of $20 \cdot \log_{10}(2^Q) = 96.33$ dB or 144.5 dB for an uniformly distributed input and $Q = 16$ or 24. Especially when working with very sensitive equipment and fixed preamplification parameters, in an environment featuring a high dynamic range, the choice of this bit depth becomes important concerning the signal to noise ratio of weak signal sources. In the further theoretical details, however, the quantising step will be neglected for the sake of simplicity and the signals will be handled as continuous-valued functions bounded by the extreme values of -1 and 1.

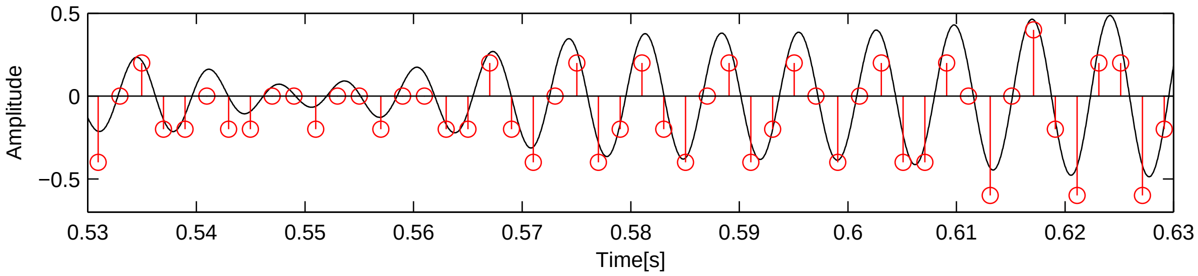
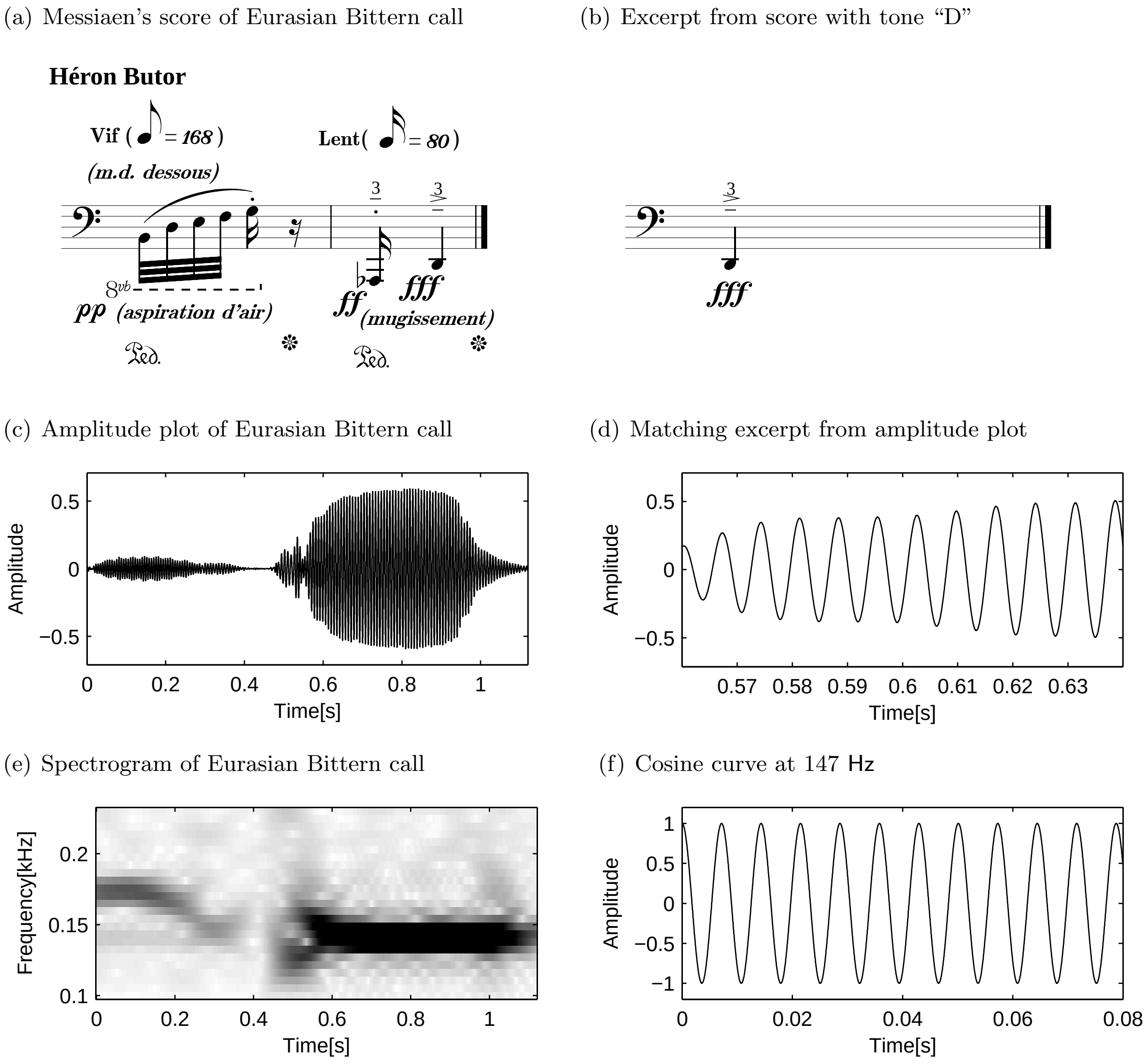


Figure 3.3: 100 ms excerpt of the Bittern signal, sampled at 500 Hz and quantised using a total of 9 quantisation intervals. The sampled values are indicated by red stems.

## 3.3 Spectral analysis and filters

Despite notable exceptions such as those proposed in [Che01], only few features meaningful for a robust pattern recognition task can be derived directly from the above amplitude-oriented signal representation. As shown in the left hand side of Figure 3.4, the amplitude plot (c) of an recorded signal is quite useful for the comparison of amplitude parameters: the pianissimo(pp) and fortissimo(ff and fff) sections noted in Messiaen's score have their obvious correspondences in the amplitude plot. However, the tonal or frequency information cannot be intuitively derived from the latter representation. In order to gain more information, especially on the



(a) Messiaen's score of Eurasian Bittern call

(b) Excerpt from score with tone "D"

(c) Amplitude plot of Eurasian Bittern call

(d) Matching excerpt from amplitude plot

(e) Spectrogram of Eurasian Bittern call

(f) Cosine curve at 147 Hz

Figure 3.4: Left column: comparison between different representations of a Bittern call: (a) score, (c) amplitude and (e) spectrogram. Right column: motivation of decomposition performed by the Fourier transform: (b) score, (d) matching excerpt from Bittern's amplitude representation, and (f) cosine function used for the frequency analysis.

frequencies contained in the input, the discrete signal is transformed into the frequency domain using a discrete windowed Fourier transform, the WFT. Therefrom we derive some elementary features by means of partitioning and coarsening the WFT data. The WFT representation is very similar to the sonogram, which was widely used by ornithologists to analyse bird voices. Both procedures can be used to analyse and display the frequency content of a sound and its variation in time, using a two-dimensional image. Here, time is represented on the horizontal axis while the vertical position of a point refers to a frequency band. The intensity of a dot in that image measures the amplitude of the associated frequency at a specific time instant. Now, comparing the spectrogram (e) to the score, some tonal differences between the artistic work in the score and the sound, analysed by means of the spectrogram, become obvious: although the last note "D" of the left-hand score, corresponding to a frequency of about 147 Hz, is matched in the spectrogram, the uprising sequence in the beginning of the score is inverted in the actual recording. Actually, the score and spectrogram depict calls from different individuals, thus representing different audio signals.

### 3.3.1 The Fourier transform

The Fourier transform originates from the Fourier series, named after the French mathematician Jean Baptiste Joseph Fourier (1768 - 1830). It defines an orthogonal transform of periodic signals, for example $f \in L^2([0,1])$, to the orthonormal basis

$$\left\{\mathbf{1}, \sqrt{2}\cos(2\pi k\cdot), \sqrt{2}\sin(2\pi k\cdot) | k \in \mathbb{N}\right\}, \tag{3.4}$$

where $\mathbf{1}$ represents the unity function defined by $\mathbf{1}(t) = 1, \forall t \in \mathbb{R}$. Since these sine and cosine waves can be interpreted as prototypes for periodic sounds of different pitches, like very primitive tones, decomposing sounds into (infinite) mixtures of the above base vectors will reveal their tonal or frequency-related properties. In fact, many birdsongs have a strong tonal character, making them accessible for imitations by flutes or whistling. The call of the Eurasian Bittern, as displayed in Figure 3.4 is one of them. As shown in the right column of this figure, the last, and loudest portion of the call (excerpt shown in (d)) corresponds to the musical note "D" (top), which in turn is usually associated to a frequency of 147 Hz, represented by the cosine curve at the bottom. Thus, the aspired decomposition of bird sounds into frequency signals promises to deliver an appropriate representation. Using Euler's theorem, the above basis can also be expressed as

$$\{e^{2\pi i k\cdot}\}, \quad \text{for } k \in \mathbb{Z}, \tag{3.5}$$

comprising the sine and cosine information in the real and imaginary parts of the exponential. For a discrete time signal $x \in \ell^2(\mathbb{Z})$, the Fourier transform (FT) is defined as follows:

$$\hat{x}(\omega) := \sum_{k=-\infty}^{\infty} x(k) e^{-2\pi i k\omega}, \quad \text{for } \omega \in [0,1]. \tag{3.6}$$

The Fourier transformed signal $\hat{x} \in L^2([0,1])$ is a $[0,1]$-periodic function having bounded energy. A coefficient $\hat{x}(\omega)$ corresponds to the average intensity of frequency $\omega$ within the analysed signal. The imaginary and complex part of a coefficient refer to two signals that are equal to each other except for a phase shift of 0.25, their ratio expressing information about the signal's phase. In this thesis, we will concentrate on the absolute value of the spectral coefficients, coinciding with the energy of the specific frequency in the signal. This representation is called the *power spectrum.* Thus, the power spectrum of a sine wave $\sin(2\pi\omega\cdot)$, oscillating at frequency $\omega$, will yield a single peak at that frequency.

Despite this intuitive analytical character, another quite practical property of the Fourier transform is given by the following fact: In the frequency domain, the convolution operator is converted to pointwise multiplication. Defining the convolution $x * y$ of two discrete time signals $x, y \in \ell^2(\mathbb{Z})$ as

$$(x * y)(n) = \sum_{k=-\infty}^{\infty} x(k)y(n-k). \tag{3.7}$$

and considering the Fourier transform as defined above, the convolution theorem states that

$$\widehat{(x * y)}(\omega) = \hat{x}(\omega) \cdot \hat{y}(\omega). \tag{3.8}$$

Thus, with the inverse Fourier transform defined as

$$\check{x}(k) := \int_0^1 x(\omega)e^{2\pi ik\omega}d\omega, \quad \text{for } k \in \mathbb{Z}, \tag{3.9}$$

one can perform the convolution of two vectors $x, y \in \mathbb{C}^N$ using both Fourier transforms: $(x * y) = \check{z}$, with $z(\omega) = \hat{x}(\omega) \cdot \hat{y}(\omega)$. As detailed below, the computational costs for one-dimensional convolution can be reduced from $O(N^2)$ to $O(N \log N)$ for two finite signals $x, y$ of length $N$.

With finite hardware, the implementation of the infinite sum and integral used in the above equations is infeasible. We will now explain a discrete approximation of the Fourier transform, which operates on finite subsequences of a discrete time signal. The discrete Fourier transform of size $N$ is an unitary isometry $\mathrm{DFT}_N : \mathbb{C}^N \to \mathbb{C}^N$. For a finite discrete signal or vector $x$ of length $N$, the DFT is defined as

$$X(j) := (DFT_N\, x)\,(j) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} x(k)e^{-2\pi i\frac{kj}{N}}, \quad \text{for } j \in \{0, 1, \ldots, N-1\}. \tag{3.10}$$

The inverse discrete Fourier transform (IDFT) is defined as

$$x(k) := (IDFT_N\, X)\,(k) = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} X(j)e^{2\pi i\frac{kj}{N}}, \quad \text{for } k \in \{0, 1, \ldots, N-1\}. \tag{3.11}$$

Thus, the j-th row of the $\mathrm{DFT}_N$-matrix, acting as a frequency reference or sampled exponential curve, is given by powers $\Omega_N^{jk}, k \in \{0, 1, \cdots, N-1\}$ of the $N$-th roots of unity

$$\Omega_N = e^{-2\pi i\frac{1}{N}}, \tag{3.12}$$

and thus

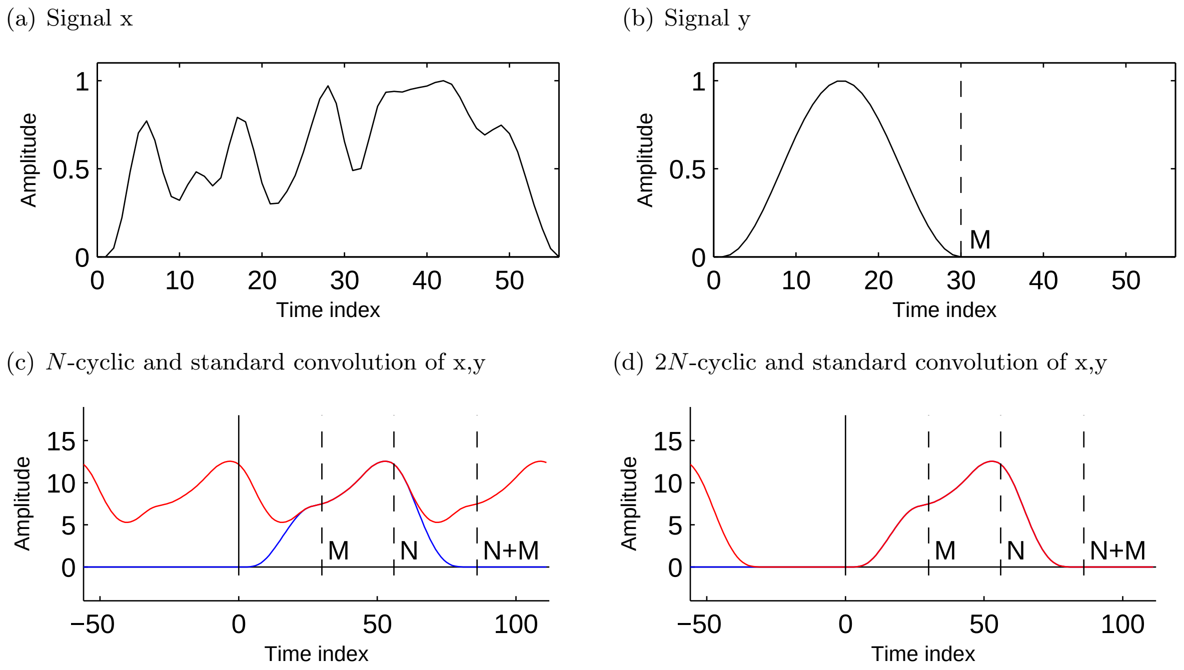
$$\mathrm{DFT}_N(j,k) \;:=\; \frac{1}{\sqrt{N}}\Omega_N^{jk}, \quad j,k \in \{0, 1, \cdots, N-1\} \tag{3.13}$$

$$\mathrm{IDFT}_N(j,k) \;:=\; \frac{1}{\sqrt{N}}\Omega_N^{-jk}, \quad j,k \in \{0, 1, \cdots, N-1\}. \tag{3.14}$$

The DFT uses a discrete and equidistant sampled set of base vectors to approximate the Fourier transform. Sampling the frequency domain implies a time domain N-periodicity of the signal to be transformed. Thus, the above convolution theorem (3.8), does not apply for the DFT. Instead of calculating the convolution of finite dimensional vectors, the convolution of two periodic discrete time signals is performed, resulting in what is called the cyclic convolution. In this, the cyclic convolution $x *_N y$ of two discrete, finite time signals $x, y \in \mathbb{C}^N$ is defined as

$$(x *_N y)(n) = \sum_{k=0}^{N-1} x(k)y((n-k) \bmod N). \tag{3.15}$$

Now, for the DFT, $(\mathrm{DFT}_N(x *_N y))\,(j) = X(j) \cdot Y(j)$, $\forall j \in \{0, \cdots, N-1\}$ holds.


(a) Signal x

(b) Signal y

(c) $N$-cyclic and standard convolution of x,y

(d) $2N$-cyclic and standard convolution of x,y

Figure 3.5: Cyclic and standard convolution of signals $x$ and $y$. The blue curve on the bottom depicts standard convolution, the above red curve plots cyclic convolution.

As the above cyclic convolution is, also, $N$-periodic, the $N$ DFT coefficients contain all the usable information. In contrast to the standard convolution, having a data length of $2N-1$ points. In the bottom left plot of Figure 3.5, the two convolution approaches are compared using two signals $x, y \in \mathbb{C}^N$, $N = 56$. Here, $y$, only having a support of length $M < N$, has been extended using zero values. The standard convolution of $x$ and $y$ is plotted as the lower, blue curve. The red curve, plotted on top of the latter, depicts the cyclic convolution of the signals. As an effect of the periodic continuation of the functions, only the positions $M \leq p \leq N$ feature identical values for both curves. As shown in the bottom right plot (d), it is also possible to calculate the standard convolution of $x, y \in \mathbb{C}^N$, as stated in (3.7), by using the $\mathrm{DFT}_{2N}$. Therefore, both signals are extended to $\mathbb{C}^{2N}$ by means of adding zero values. E.g. $x$ is extended to

$$x'(n) \quad = \quad \begin{cases} x(n) & \text{if } 0 \leq n < N, \\ 0 & \text{otherwise.} \end{cases} \tag{3.16}$$

Now, the $\text{DFT}_{2N}$ is used to calculate $(x' * y')$. We are interested in the convolution coefficients $0 \leq n < 2N$. The periodically continued signals have zero values at the last $N$ positions. Thus, by doubling the convolution's period, the overlapping parts only contain zeros, and we arrive at the standard convolution operation.

Although calculating the double amount of data points seems inefficient at first glance, it leads to a very efficient algorithm for fast convolution: being expressed as an $N \times N$-matrix, with $N = 2^n, n \in \mathbb{N}$, calculating the DFT is possible in time $O(N \log N)$ by using a matrix factorisation-based algorithm rediscovered by Cooley and Tuckey. This algorithm is also known as the fast Fourier transform (FFT). Thus, combining the previous statements, the cyclic convolution $x *_N y$, can be performed by computing three $O(N \log N)$ transformations instead of needing $O(N^2)$ time.

In the further explanations, a two-dimensional convolution and FFT algorithm will be used for some image processing routines. For $x, y \in \mathbb{C}^{N \times M}$, the convolution operator $x * y$ is defined as

$$(x * y)(n, m) := \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} x(k, l) y(n - k, m - l). \tag{3.17}$$

As for the one-dimensional case, the convolution can be efficiently computed using the DFT for the zero-padded signals in $\mathbb{C}^{2N \times 2M}$. For $x \in \mathbb{C}^{N \times M}$, the two-dimensional $DFT_{N \times M}$ is defined as

$$X(n, m) := \frac{1}{\sqrt{NM}} \sum_{k=0}^{N-1} \Omega_N^{kn} \sum_{l=0}^{M-1} x(k, l) \Omega_M^{lm}, \tag{3.18}$$

where $n \in \{0, 1, \ldots, N-1\}, m \in \{0, 1, \ldots, M-1\}$. The above formula can be obtained by using two one-dimensional DFT's as follows, using matrix notation:

$$X^\top = \text{DFT}_M (\text{DFT}_N \, x)^\top. \tag{3.19}$$

### 3.3.2 Digital filters

In the following, linear systems $T : \ell^2(\mathbb{Z}) \to \ell^2(\mathbb{Z})$ will be used to "smooth" various kinds of signals. Considering the particular systems used in this work, we focus our attention on the class of Finite Impulse Response (FIR) filters. Given a signal $x \in \ell^2(\mathbb{Z})$, an FIR filter $T$ is defined through the convolution with an associated signal $h_T \in \ell^1(\mathbb{Z})$, having a limited number of non-zero coefficients:

$$T[x](k) := (h_T * x)(k) \in \ell^2(\mathbb{Z}) \tag{3.20}$$

Thus completely defining the filter function, $h_T = T[\delta]$, where

$$\ell^1(\mathbb{Z}) \ni \delta(t) := \begin{cases} 1 & \text{for } t = 0 \\ 0 & \text{otherwise,} \end{cases}$$

$h_T$ is called the *impulse response* of the filter. As the convolution operator is linear, the filter function adopts this property. Furthermore, the FIR filters are time invariant: $T[x(\cdot + c)](t) = T[x](t+c)$. Because $h_T$ only has a finite number of relevant coefficients, the length of an FIR filter may be defined as

$$l(T) := 1 + \max\{n \mid h_T(n) \neq 0\} - \min\{n \mid h_T(n) \neq 0\}. \tag{3.21}$$

More generally, for $h_T \in \ell^1(\mathbb{Z})$ of finite length and $x \in \ell^p(\mathbb{Z})$, the filter output $T[x] \in \ell^p(\mathbb{Z})$, is defined and, considering the Young inequality, bounded by $\|T[x]\|_p \leq \|h_T\|_1 \cdot \|x\|_p$. In short, FIR filters are stable LTI (linear time invariant) systems.

Actually, dealing with finite signals $x, h_T \in \mathbb{C}^N$, the length of a filtered signal is preserved by restricting the calculated convolution coefficients to $n \in \{0, \cdots, N-1\}$. For example, in the previous Figure 3.5, a signal $x$ is convolved with a Hann window (signal $y$). The signal plotted in the bottom right plot, cut down to the indexes $n \in \{0, \cdots, N-1\}$, represents the result of the filter $T[x](n)$, for $h_T := y$.

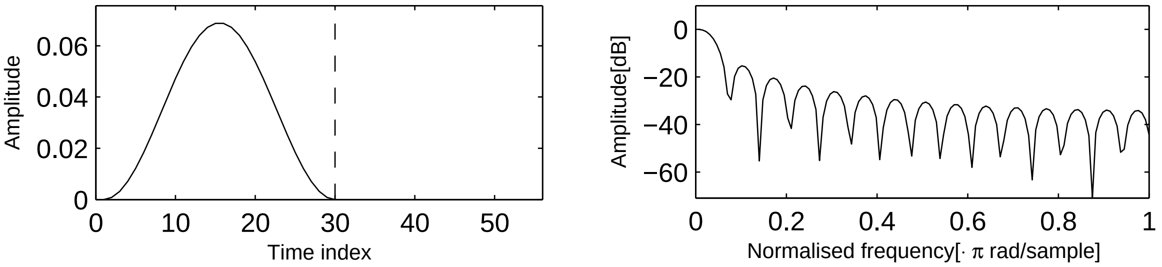


Figure 3.6: Left: amplitude $h_T$ and Right: power spectrum $\|H_T\|$ plot of a scaled Hann window. The Fourier transformed coefficients are depicted on a logarithmic scale.

In the frequency domain, FIR filters are represented by the Fourier transform of their impulse response. In Figure 3.6, a Hann window $h_T$ and its power spectrum $\|H_T\|$ are depicted. Examining the logarithmically plotted spectral energy distribution, the coefficients' values decrease with increasing frequency. Now we recall, that the convolution performed when filtering can be expressed as pointwise multiplication of $X$ and $H_T$ in the frequency domain. As depicted in the latter figure, most of the high-frequency contents of a Hann-filtered signal will be damped. This causes the desired smoothing of the filtered signal. Moreover, the filtering process has also influence on the phase of the signal's components. Thus, the temporal structure of a filtered signal may vary significantly from the original. Given the smoothing applications of the Hann window to be discussed in this work, the magnitude of this displacement is sometimes neglected in a tradeoff regarding computation time and temporal accuracy.

Another FIR filter type frequently used in the following algorithms is the sliding mean or averaging filter. Here, the basic idea is to replace each data point by a mean value of its neighbourhood. Limiting the temporal range of this comparison, a sliding mean filter $T_s$, considering a neighbourhood of $s$ signal coefficients, may be expressed by its impulse response:

$$h_{T_s} := (1, \cdots, 1) \cdot \frac{1}{s} \quad \in \mathbb{C}^s \tag{3.22}$$

In the further text, an adapted version of this filter will be applied on a signal $x \in \mathbb{C}^N$. Note, that the following filter is not time-invariant. Let $s$ be sufficiently large and $s_h := \lfloor s/2 \rfloor$, then, considering the intention of a moving average, a more suitable solution will be referred to as

$$\overset{s}{\bar{x}}(n) := \begin{cases} T_s[x](n+s_h)\cdot \frac{s}{s-(s_h+n)} & 0 \leq n < s_h \\ T_s[x](n+s_h) & s_h \leq n < N - s_h \\ T_s[x](n+s_h)\cdot \frac{s}{N-(n-s_h)} & N - s_h \leq n < N \end{cases} \tag{3.23}$$

Here, an approximative lag of $s_h$ samples is reversed by time-shifting the whole averaged sequence. Furthermore, the edge-effects of decreasing values are nearly balanced.

### 3.3.3 WFT and spectrogram

As the classical Fourier transform analyses the whole input signal, it yields mean values of the included spectral components. Due to the importance of temporal variation in birdsongs, we are however interested in a more localised representation. The basic idea of the windowed Fourier transform is to break the signal into small, possibly overlapping short-time frames, which are then analysed separately. These excerpts are isolated by a windowing function $g(t-t_0)$. Usually, $g$ represents a real-valued, even function centred at $t = 0$, that suppresses the signal $x$ outside a certain region, when $x$ is multiplied by $g$. Let $g \in \ell^2(\mathbb{Z}), \|g\|_2 \neq 0$, be a window function with finite, non zero energy. For $x \in \ell^2(\mathbb{Z})$, the WFT is then defined as the scalar product, defined in (3.3), of $x$ and the windowed frequency signals $g_{\omega,t_0}(t) := e^{2\pi i\omega t}g(t-t_0)$:

$$\widetilde{x}(\omega, t_0) = \langle x, g_{\omega,t_0}\rangle\,, \quad \text{for } \omega \in [0,1]. \tag{3.24}$$

In practical applications, (3.24) is transferred to a discrete-frequency version. Let $g \in \mathbb{C}^N$ denote a windowing vector, representing a windowing function having a support of $s_w = N$ successive nonzero coefficients. Then, $x_{t_0} \in \mathbb{C}^N$ is defined as a vector containing the remaining coefficients of the pointwise multiplication of $x$ and $g(t-t_0)$,

$$x_{t_0} := x(t)g(t-t_0), \quad \text{for } t \in \{t_0+p, t_0+(p+1), \cdots, t_0+q\}, \tag{3.25}$$

with

$$\begin{aligned} p &= \min\{t \in \mathbb{Z} \mid g(t) \neq 0\}, \\ q &= \max\{t \in \mathbb{Z} \mid g(t) \neq 0\}, \end{aligned}$$

and $q - p = N - 1$. Now, with $t_0 \in \mathbb{Z}$ and $j \in \{0, 1, \cdots, N-1\}$, the DWFT is defined as the scalar product of the windowed signal and the sampled frequency signals:

$$\widetilde{x}(j, t_0) = \left\langle x_{t_0}, \mathrm{DFT}_N(j,\cdot)^\top \right\rangle. \tag{3.26}$$

In order to derive the desired spectrogram, the WFT is computed for equidistantly spaced WFT-frame positions $t_0 \in \{t_s + n\cdot\Delta_w \mid n \in \mathbb{N}_0\}$, where $\Delta_w$ determines the *step size* and thus the overlapping range of two successive windows (see Fig. 3.7). $t_s$ refers to the first window position. As a matter of fact, both the time and frequency resolution of the previous transform are influenced by the window's support's length $N$ in a competitive manner: as $N$ grows, the frequency domain is represented more precisely due to the denser sampling, while there are more time domain samples contributing to a spectral coefficient. This has a negative effect on time resolution. The windowing function's shape is determining the above parameters, too. According to the Heisenberg Uncertainty Principle, the Gaussian bell functions come with an optimal tradeoff between localisation in the frequency and time domains, when used as windowing functions. As the Gaussian window function has infinite support, suitable approximations as well as more or less similar functions are used for practical purposes. In this thesis, a Hann window function is used:

$$g(n) := \begin{cases} 0.5 \cdot (1 - \cos(2\pi \frac{n}{s_w+1})), & \text{for } 1 \leq n \leq s_w, \\ 0, & \text{otherwise.} \end{cases} \tag{3.27}$$

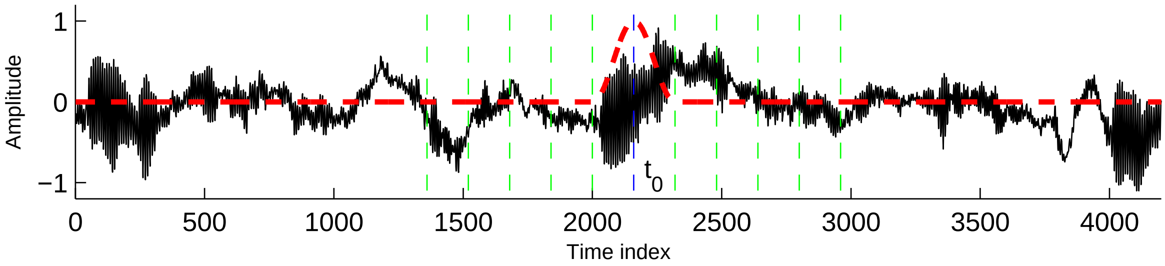


Figure 3.7: Amplitude plot of an 87 ms excerpt from a Grasshopper Warbler's (*Locustella naevia*) song. Hann window centered at $t_0$ plotted as red, dashed curve. Some further WFT-frame positions indicated by vertical lines.

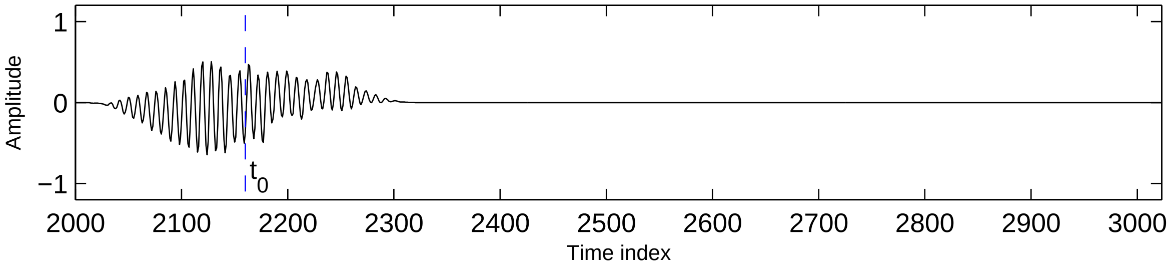


Figure 3.8: 1024 samples excerpt of windowed signal from Figure 3.7.

The stepwidth $\Delta_w = \frac{s_w}{2}$ is set to half a window's width to achieve an overlap of the same length. Although the number of DFT coefficients is fixed at $N = 2048$ or 1024 samples, the Hann window's width $s_w$ and thus the excerpt of the input signal is allowed to be smaller than $N$. In this way, components beyond a certain time position are neglected and the signal is zero padded to a length of $N$ coefficients. Thus, retaining $N$, one can control the time resolution of the spectrogram by setting the window's width. For example, the Savi's Warbler detector analyses only 320 samples per WFT-frame (see Fig. 3.8), corresponding to a frame rate of fps = 300 frames per second, whereas the low time resolution achieved by using the full DFT size $s_w = N$, corresponding to 50 frames per second, is sufficient for the detection of more slowly developing melodies like in the Chaffinch's song.

Since we are only interested in the magnitude of each spectral component, the squared absolute values are calculated for each complex spectral coefficient. The phase information is discarded. This representation is also called the power spectrum. Because of the almost logarithmic magnitude perception performed by the human ear and in order to get a more dense energy distribution, the natural logarithm is applied to the magnitude values. Thus, the influence of a volume modulation decreases with the respective amplitude. We obtain a modified spectrogram

$$\mathrm{specgr}^*(j,k) := \log\left(\|\widetilde{x}(j, t_s + k \cdot \Delta_w)\|^2 + 10^{-8}\right),$$

which is finally, for reasons of numerical stability concerning the further process, normalised to a maximum value of 1, by using a factor $c = \max\limits_{j,k}(|\mathrm{specgr}^*(j,k)|)$,:

$$\mathrm{specgr}(j,k) := 1 + \frac{\mathrm{specgr}^*(j,k)}{c}. \tag{3.28}$$

# Chapter 4

# Acoustical features

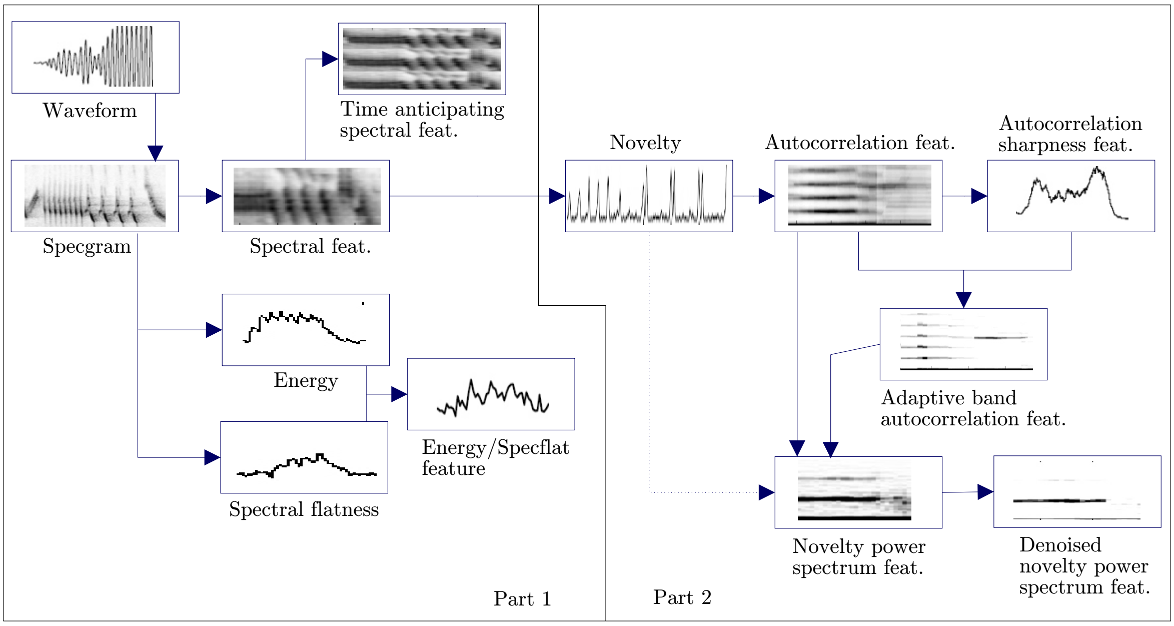

Waveform
Time anticipating spectral feat.
Specgram
Spectral feat.
Energy
Energy/Specflat feature
Spectral flatness
Part 1
Novelty
Autocorrelation feat.
Autocorrelation sharpness feat.
Adaptive band autocorrelation feat.
Novelty power spectrum feat.
Denoised novelty power spectrum feat.
Part 2

Figure 4.1: Chart of feature dependencies.

As discussed in Chapter 2.3, the spectrogram derived above, facilitating a very rich depiction of acoustic signals, is now used as a basis for the modelling of further representations. In the following sections, several features will be defined, each measuring an individual aspect of the acoustic signal. As the focus of this thesis is on birdsong recognition, the envisaged features are designed to robustly measure parameters being characteristic for this task, while being invariant up to noise and other irrelevant sounds.

The construction of these features, as depicted in the flow chart shown in Figure 4.1, is divided in two parts: Firstly, the spectrogram is analysed by means of energy and shape measurements. Furthermore, a coarsened, but also robust representation of the spectrogram is derived. Moreover, the latter spectral features are concatenated in order to encode some temporal information.

In the second part, discussed in section 4.2, the introduction of novelty curves permits the extraction of the even more robust, but also more specialized periodicity features. Here, repetitive structures are extracted from the spectrogram, forcing the definition of a second order frequency: the element repetition frequency. By means of the (adaptive) autocorrelation features, a quite robust representation of periodic acoustic events can be extracted. Frequency-transforming the previous features, the novelty power spectrum features allow for an additional denoising method. As each of the latter representations has been developed for an individual application, the detectors, build upon the basic features established in the following paragraphs, are meant to associate each feature with its respective scenarios.

## 4.1 Frequency bands and related features

For the applications described in this work, we will now focus our attention to one or more frequency bands $B_0, \cdots, B_{N-1}$. For many bird identification tasks, it is useful to focus on a distinct frequency band, covering most of the frequency range which is used by the bird. In this case, we only need a single band located between the indexes $p_l, q_l$ for $l = 0$, defining the bands minimum and maximum frequencies. These variables are set manually according to some experimentally or statistically derived knowledge. In the following, the term *frequency band* or band will refer to the set of spectral coefficients $B_l := \{j \mid p_l \leq j \leq q_l \in \mathbb{N}_0\}$ belonging to the chosen frequency range with limiting indices $p_l, q_l$. The first audio feature proposed in this work measures a value proportional to the signal energy contained in a whole band. This *band energy* feature is computed by frame-wise adding the assigned coefficients.

$$\text{energy}[B_l](k) := \sum_{j \in B_l} \text{specgr}(j, k). \tag{4.1}$$

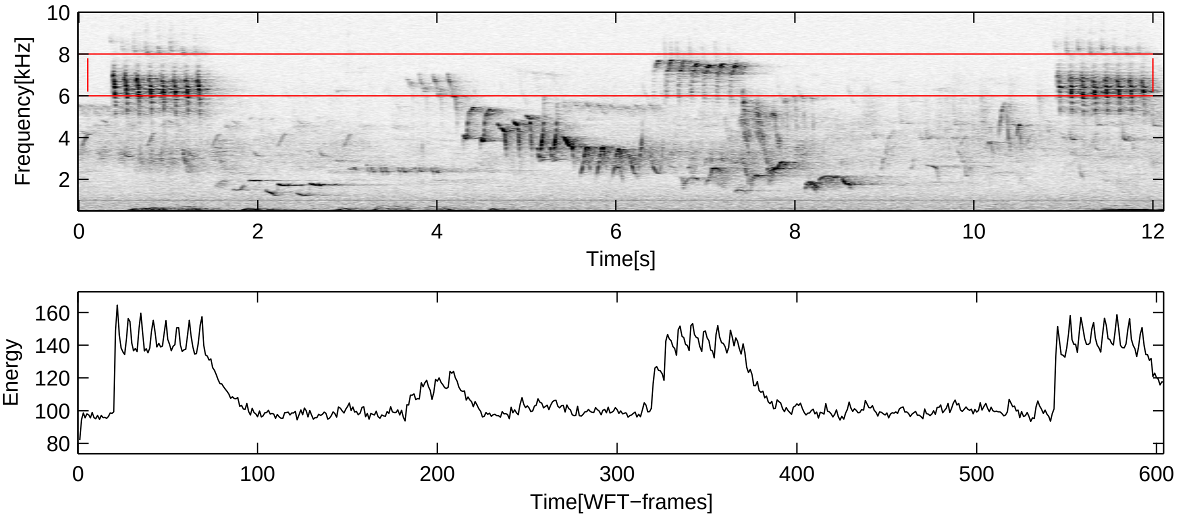


Figure 4.2: 6-8 kHz subband of a monitoring recording. Top: spectrogram. Bottom: energy plot as defined in equation (4.1).

Monitoring the band energy over time, one can detect sequences of high energy which may indicate a significant event. Besides being used to generate a noise level estimate as described in [Fag04], the band energy feature may be combined with another standard signal processing feature: the *spectral flatness* measure serves as an indicator for a homogeneous energy distribution among the spectral coefficients. In general, the positive valued flatness measure is given by dividing the geometric energy mean

$$\breve{B}_l := \sqrt[|B_l|]{\prod_{j \in B_l} \operatorname{specgr}(j,k)}$$

by the arithmetic mean

$$\overline{B_l} := \frac{\operatorname{energy}[B_l](k)}{|B_l|},$$

$$\operatorname{specflat}[B_l](k) := \frac{\breve{B}_l}{\overline{B_l}} \leq 1. \tag{4.2}$$

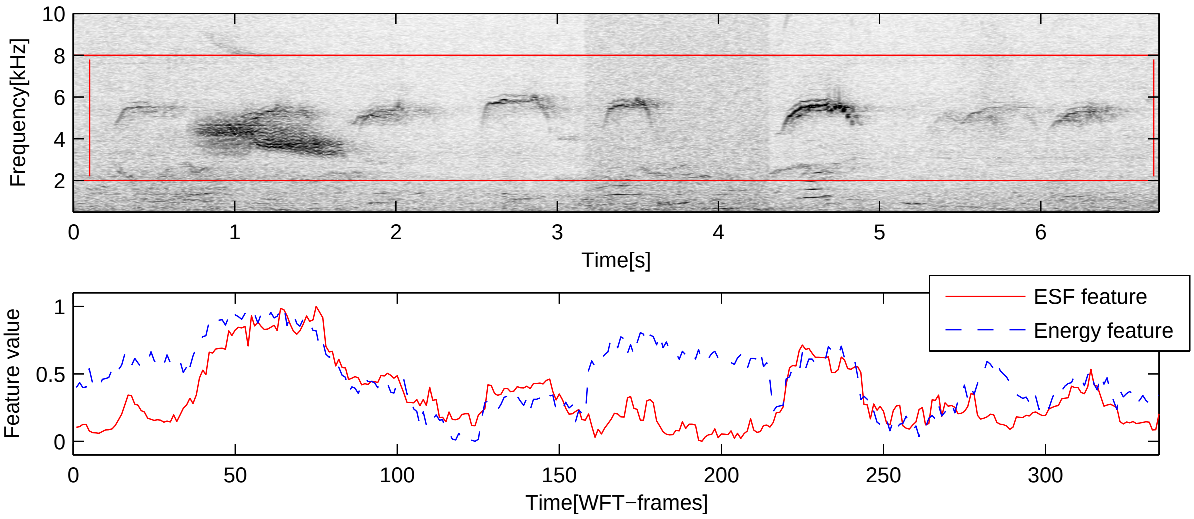


Figure 4.3: Top: Spectrogram of a monitoring recording, with framed frequency band from 2-8 kHz, containing calls of the Common Swift (*Apus apus*). Bottom: comparison of shapes of energy and ESF features.

Here, $|B_l|$ counts the spectral coefficients associated to the frequency band $B_l$. Unfortunately, the spectral flatness measure is very sensitive to outlying spectrogram values very close to zero. In order to avoid the resulting feature outliers, some median filtering or other smoothing should be performed on the specflat curve. The latter two features may be combined to build an $\frac{\text{energy}}{\text{specflat}}$ (ESF) feature indicating high overall energy concentrated in only a few coefficients. Unfortunately, spectral flatness values are somewhat likely to be zero, and thus the above definition fails. A more robust but similar ESF measure can be calculated as follows:

$$F_{\text{ESF}}[B_l](k) := \frac{\operatorname{energy}[B_l](k)}{|B_l|} \cdot (1 - \operatorname{specflat}[B_l](k)). \tag{4.3}$$

In this definition, the second term measures the roughness of a spectrogram vector. The ESF feature is particularly useful for the detection of song birds, because many of their songs contain sine-like tones, which usually results in a significant ESF peak. In Figure 4.3, the shapes of both the energy and ESF measurements are compared, using a signal with alternating noise levels and sinusoidial bird calls. Both curves are normalised in order to ease the comparison. In the middle part of the excerpt, some synthetic noise was added, which, being equally distributed on all frequency bins, is only reflected by the energy measure.

### 4.1.1 Spectral features

In order to smooth out noise in the spectrogram, it is filtered with a two dimensional Gaussian. The continuous Gaussian Hat signal is defined as

$$\operatorname{gauss}(p,q) := \frac{1}{2\pi} \cdot e^{-\frac{1}{2}(p^2+q^2)}, \quad \text{for}\, p,q \in \mathbb{R}.$$

We use a sampled version, gaining a FIR filter with an impulse response displayed in the following matrix:

$$\begin{pmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 8 & 11 & 8 & 2 \\ 3 & 11 & 15 & 11 & 3 \\ 2 & 8 & 11 & 8 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{pmatrix} \tag{4.4}$$

The filtering is done using the fast convolution algorithm for image signals, described in Chapter 3.3. Thus, simultaneously smoothing the spectrogram in the time and in the frequency domain, grainy noise peaks are flattened out. We now focus on the frequency band which is used by the bird. Therefore, the coefficients which do not contribute to the specific band are discarded at this stage. Afterwards, the frequency resolution is reduced by pooling spectral coefficients into $\#_{\text{bins}}$ bins. This is done by triangularly weighted summation of successive coefficients, which can be interpreted as a windowing procedure in the frequency domain. For $n \in \mathbb{N}$, the triangular windows of width $s_b$ are defined as zero-padded vectors $\mathrm{WIN}_T \in \mathbb{C}^N$:

$$\mathrm{WIN}_T(n) := \begin{cases} \frac{2n}{s_b+1} & \text{if } 1 \le n \le \frac{s_b+1}{2}, \\ \frac{2(s_b-n+1)}{s_b+1} & \text{if } \frac{s_b+1}{2} < n \le s_b, \\ 0 & \text{otherwise, and} \end{cases} \quad \text{for}\, s_b \text{ even} \tag{4.5}$$

$$\mathrm{WIN}_T(n) := \begin{cases} \frac{2n}{s_b} & \text{if } 1 \le n \le \frac{s_b+1}{2}, \\ \frac{2(s_b-n+1)}{s_b} & \text{if } \frac{s_b}{2}+1 \le n \le s_b, \\ 0 & \text{otherwise.} \end{cases} \quad \text{for}\, s_b \text{ odd}$$

During the binning process, the window function is arranged at linearly increasing positions $f_0 \in \{f_s + n\Delta_b \mid 0 \le n < \#_{\text{bins}}\}$. The individual bins have an equal number of coefficients $s_b = \left\lfloor \frac{|B_L|}{\#_{\text{bins}}} \right\rfloor$ contributing to their value. Here, $f_s$ denotes the frequency band's absolute offset.

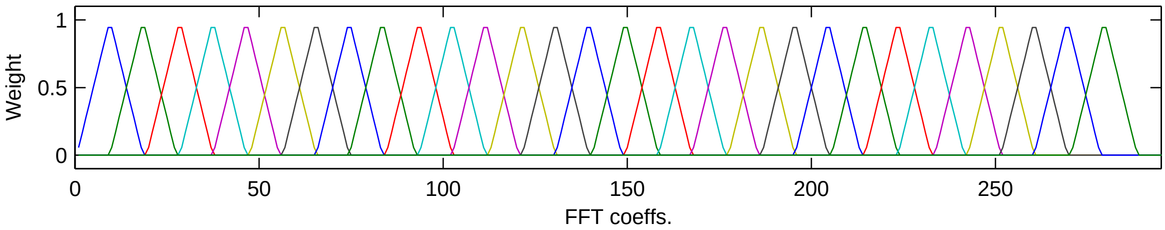


Figure 4.4: Illustration of all triangular windows used in the binning process. Each bin is associated to a single triangular window.

A stepwidth $\Delta_b$ is chosen to achieve an overlap of approximately $\frac{s_b}{2}$ frames. The weighted spectrogram coefficients are then summed up and saved in the spectral feature vector:

$$F_{\mathrm{SPEC}}(j,n) := \langle \mathrm{specgr}(\cdot,n), \mathrm{WIN}_T(\cdot - (f_s + j\Delta_b)) \rangle \, , \tag{4.6}$$

for $n \in \mathbb{Z}$ and $j \in \{0, \cdots, \#_{\mathrm{bins}} - 1\}$. Coarsening the WFT representation of an audio signal as described above, we gain a suitable feature space. Usually, 30 to 40 bins are sufficient to accomplish an adequate representation of a birdsong having a bandwidth of 3 kHz. Due to the smoothing and binning operations, these features are strongly invariant considering fine noise peaks and small frequency or amplitude variations, while representing the trend of more continuous signals.

This representation can easily be enriched with information about the signal's temporal evolution. Incorporating the derivative of the single spectral bins over time is an easy way to achieve such information, concerning two successive WFT-frames. In this work, some temporal information is added to the feature vectors by simply concatenating $\#_{\mathrm{anti}} + 1$ successive feature vectors:

$$F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}} := \begin{pmatrix} F_{\mathrm{SPEC}}(0,k) \\ F_{\mathrm{SPEC}}(1,k) \\ \vdots \\ F_{\mathrm{SPEC}}(n,k) \\ F_{\mathrm{SPEC}}(0,k+1) \\ F_{\mathrm{SPEC}}(1,k+1) \\ \vdots \\ F_{\mathrm{SPEC}}(n,k+\#_{\mathrm{anti}}) \end{pmatrix}, \text{ for } \; k \in \mathbb{Z} \tag{4.7}$$

## 4.2 Periodicity features

Although the $F_{\mathrm{SPEC}}$ and $F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}}$ features contain dense information about the analysed signal, it is quite difficult to separate the signal contents from noise. Now concentrating on periodically reoccurring signals, we will propose a set of features which strongly suppress background noise as well as nonperiodic signals while revealing the parameters of periodic signal sources. Here, in contrast to the periodicity of the sine waves used for the Fourier transform, the term "periodic" refers to the repetition of acoustic patterns within a time scale of approximately

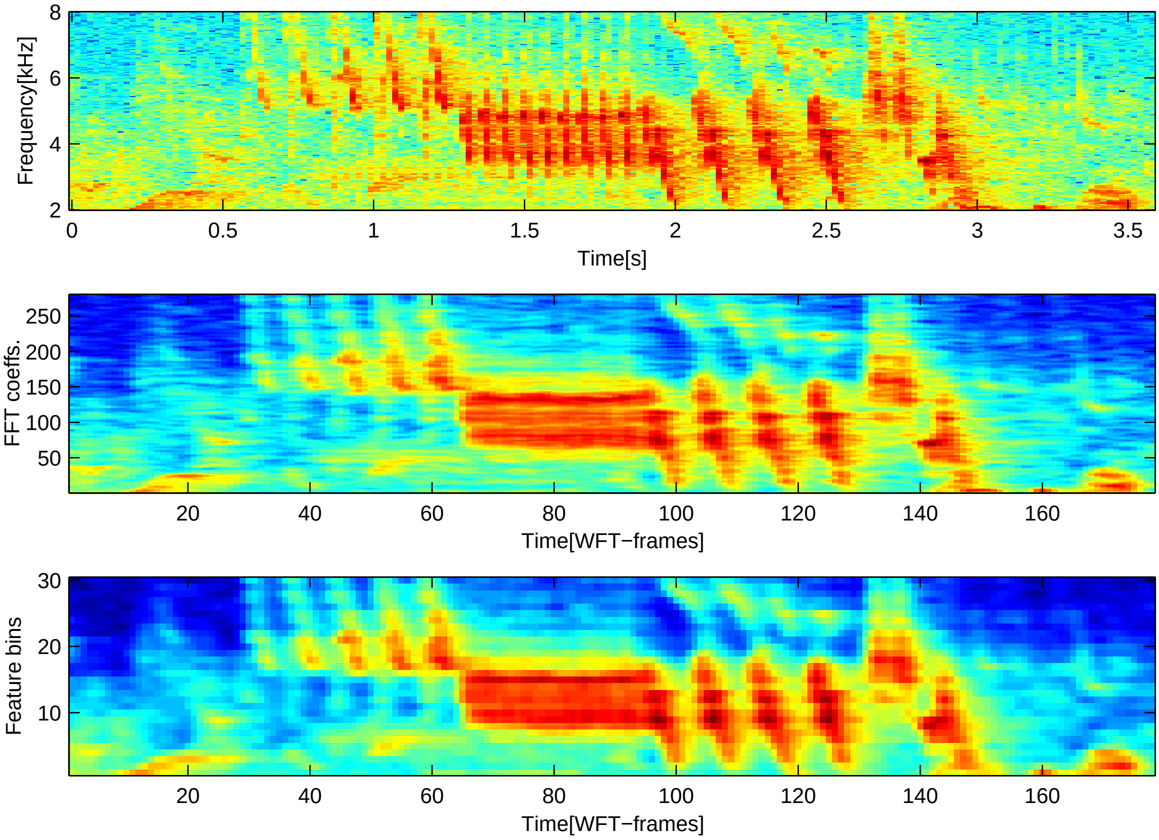


Figure 4.5: Top: 50 fps spectrogram of noisy Chaffinch stanza, frequency band: 4-8 kHz. Centre: gaussian smoothed spectrogram. Bottom: frequency binned $F_{\mathrm{SPEC}}$ features.

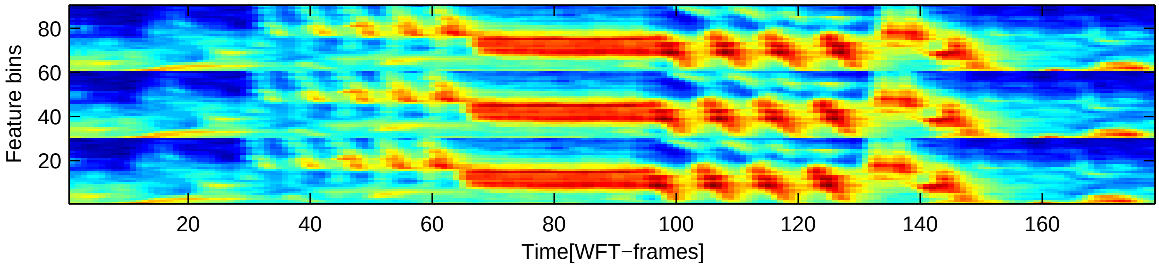


Figure 4.6: Spectral features from example shown in fig 4.5 with additional time anticipation.

25-250 ms. For the bird song identification task, in particular, the measuring of such periodic elements reveals a set of useful features. The key tool used in the feature acquisition step is autocorrelation. While the two dimensional cross-correlation measure is getting more popular for comparing spectrogram excerpts, e.g. in the Avisoft SASlab and XBAT software, we use an one dimensional autocorrelation measure comparing time-series for reasons of efficiency and robustness, which will be discussed in detail in section 4.2.2.

### 4.2.1 Novelty curves

Reducing the multidimensional $F_{\text{SPEC}}$ features to a single scalar curve often involves discarding information. Hence, the mapping used for this step has to be chosen carefully. The method proposed below retains enough relevant information while discarding nondescriptive parameters. The selection of an appropriate frequency band (see section 4.1) hence constitutes an important part of this reduction process. As we are interested in the detection of repeated elements in birdsongs, we will now focus on their limits: the onset and offset events. To this end, we now extract a curve indicating the likelihood of these events. For each feature vector $F_{\text{SPEC}}(\cdot, n)$, the amount of introduced novel information is indicated by a corresponding novelty value. Measuring the strength of spectral variation over time, the novelty curve, plotted as blue curve in Figure 4.7, is a useful indicator for on- and offset events. A common novelty measure is defined as follows:

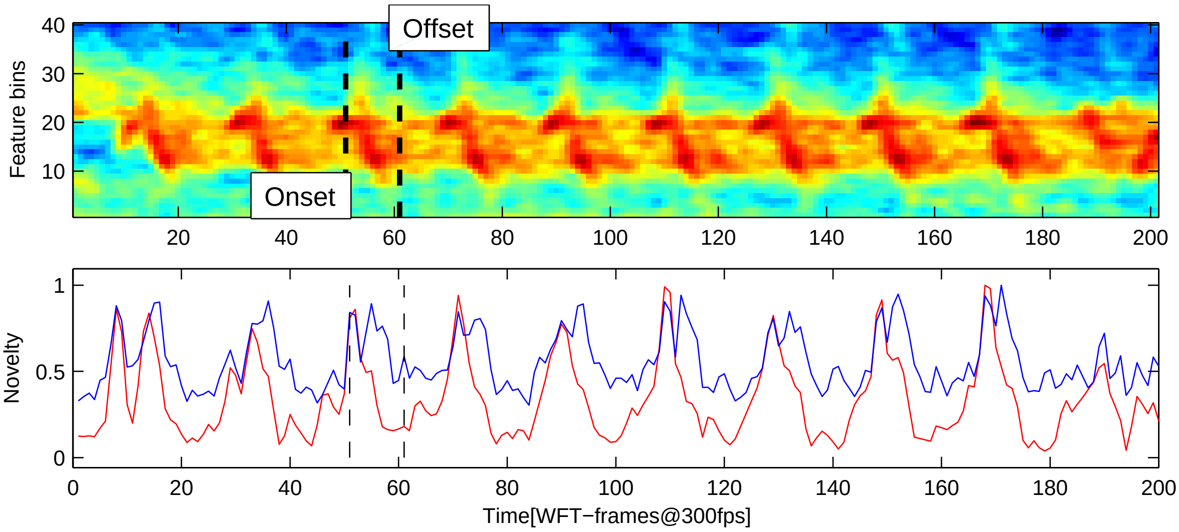


Figure 4.7: Top: excerpt from Chaffinch stanza (fig 4.5, page 46), using spectral features with high time resolution (300 frames per second). Bottom: two novelty curves. Blue common novelty measure on top of red $F_{\text{NOV}\uparrow}$ measure.

$$\text{novelty}(k) := \sum_{j=0}^{\#_{\text{bins}}-1} |F_{\text{SPEC}}(j, k+1) - F_{\text{SPEC}}(j, k)|. \tag{4.8}$$

As this curve will be used to detect the periodic repetition of song elements, we want to restrict our attention to just one type of events for the following reasons: many elements in bird sounds resemble short pulses or nearly vertical lines in the spectrum. The recordings of such signals are likely to produce at least two novelty maxima, corresponding to the onset and offset of the sounds. Unfortunately, in outdoor monitoring recordings, the energy decrease, corresponding to the offset of an emitted sound, is likely to be blurred by echo and reverb effects naturally occurring in this scenario. In the spectrogram shown in Figure 4.7, these effects fill the spaces between the single elements. In this work we will hence use a modified novelty measure which solely reflects the events corresponding to increasing energy. Thus we get maxima for the onsets of the mentioned sounds (see Fig. 4.7), by defining

$$F_{\mathrm{NOV}\uparrow}(k) := \sum_{j=0}^{\#_{\mathrm{bins}}-1} \max(F_{\mathrm{SPEC}}(j,k+1) - F_{\mathrm{SPEC}}(j,k), 0). \tag{4.9}$$

Many birdsongs contain phrases of relatively long elements resembling short sine sweeps. Usually, these sounds result in plateau-like novelty curves. This is because of the constant energy increase in the coefficients corresponding to a monotonically in- or decreasing frequency. Given a scenario in which the frequency range of a searched-after call is known and fixed, it is useful to analyse a small band $B_l$, centred at the upper or lower bound of the songs frequency range. This usually leads to more distinct novelty peaks. In some of the applications considered in the subsequent chapters, the novelty curves are calculated for a set of even smaller subbands $S(B_l)_0, \cdots, S(B_l)_M$ by subdividing the frequency band $B_l$, to minimize the plateau's width and hence to sharpen the novelty maxima. In Figure 4.8, $\#_{\mathrm{sbands}} = 9$ subbands covering a 4 kHz frequency range are extracted, gathering about 8 spectral coefficients per subband, and choosing $\#_{\mathrm{bins}} = 40$. The center frequencies of the subbands are arranged linearly and the whole bands frequency range is covered by the set. Because of their small bandwidth, the subbands are only contributing to a subsequence of a full bandwidth sweep-like sound and therefore are likely to contain shorter novelty plateaus.

The onset novelty curve explained above yields robust features which constitute the basis to the further processing step. Especially the invariance up to mid- and long scale amplitude trends is of major importance for the overall detection robustness. As the novelty features are quite sensitive to fine spectral noise, the gaussian filtered $F_{\mathrm{SPEC}}$ features are well-suited as a basis for this extraction process.

### 4.2.2 Autocorrelation

Finally, the signal is tested for periodic events. Considering the novelty curve, these events are represented by a train of equidistantly spaced local maxima. As the period of these maxima is not known a priori, we have to check for a set of relevant period values. This is achieved by the use of the statistical autocorrelation measure, a tool frequently applied in signal processing which measures the cross-correlation of a signal with itself. For $x \in \ell^2(\mathbb{Z})$, the autocorrelation measure can be easily defined as

$$\mathrm{acorr}_x(\tau) := \sum_{k\in\mathbb{Z}} x(k) \cdot x(k+\tau), \quad \text{for } \tau \in \mathbb{Z}. \tag{4.10}$$

This process might be depicted as follows: the signal $x$ is copied to $x'$, and both signals are arranged on two parallel timed tracks. To test for the zero-time ($\tau = 0$) periodicity, the

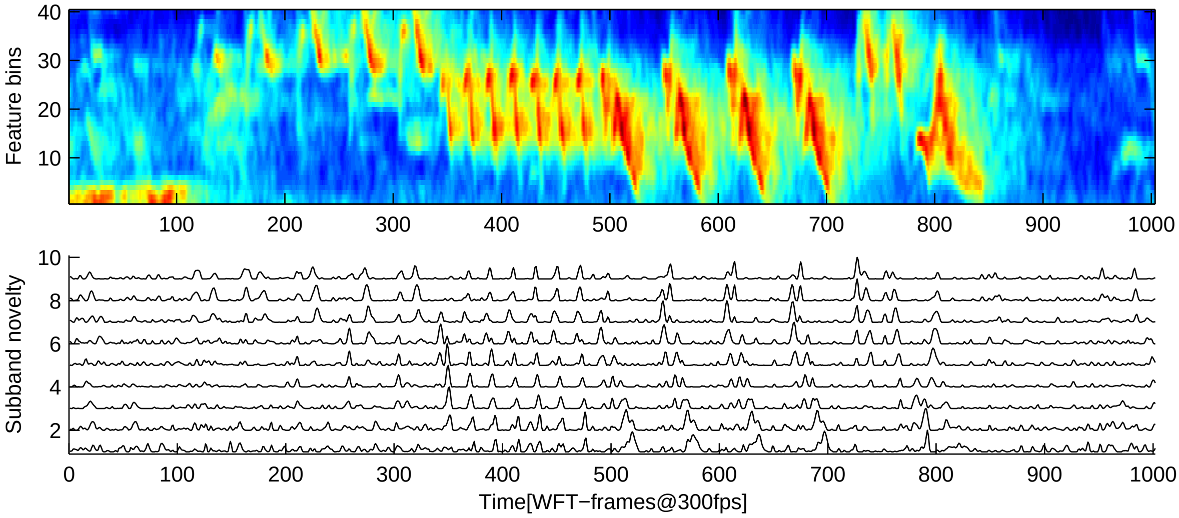


Figure 4.8: Top: spectral features of a Chaffinch's stanza. Bottom: $F_{\mathrm{NOV}\uparrow}$ novelty curves for 9 overlapping subbands. The curves' ranges have been individually normalised for better visibility.

two sequences are compared in a pointwise manner: Each pair of temporally corresponding (same horizontal position) novelty values is multiplied. The sum of the former products thus serves as a similarity measure. Now, the signal copy $x'$ is WFT-frame wise time-shifted, and the above similarity measure is successively calculated for all interesting shifts or lags $\tau$. A high $\mathrm{acorr}_x(\tau)$-value indicates a high self-similarity of the signal, and thus an approximate self-repetition of the signal within a lag of $\tau$. The above measure can be mathematically expressed and computed by convolving the signal with a reversed copy of it.

$$\mathrm{acorr}_x(\tau) = (x(\cdot) * x(-\cdot))(\tau) \tag{4.11}$$

Due to this fact, it is also possible to replace the convolution operation by a scalar multiplication in the frequency domain (see equation (3.8) and following). As we are interested in a more localized autocorrelation measure, depicting periodicities within a 300 ms range, a window based short-time autocorrelation procedure is used. This short-time autocorrelation is applied on successive, windowed excerpts of the input signal. Here, we use a rectangular window of width $s_{ac}$, corresponding to signal excerpts of sizes between 10 ms and 500 ms. Depending on the required temporal resolution, the step size $\Delta_{ac}$, interleaving the center positions of two successive windows, is chosen to obtain approximately $15 \leq \mathrm{fps}_{\mathrm{ac}} < 320$ autocorrelation vectors per second. In the following, these will be called *autocorrelation sequence*. Obviously, the windowed signals $x \in \mathbb{R}^N$, for $N = s_{ac}$, are time-limited, which turns out to be problematic. As the signal is time shifted during the autocorrelation process, we have to expand the copied signal excerpt in order to derive correlation values beyond $\tau = 0$. Here, additional coefficients $x(n)$, for $N \leq n < 2N$, are needed. In Figure 4.9, three different approaches are depicted: zero padding a signal copy $x' \in \mathbb{R}^{2N}$, as described in (3.16), comes with disadvantages as a continuous fall off of the values that corresponds to larger autocorrelation lags (see bottom blue curve). Another method is to expand the window size $s_n$ for the signal copy, thus gaining more information at the cost of temporal focus, as shown in the black, dashed curve. The autocorrelation is calculated using

$$\operatorname{acorr}'_x(\tau) := \sum_{k=0}^{N-1} x(k) \cdot x'(k+\tau), \quad \text{for } \tau \in \{0, 1, \cdots, N-1\} \tag{4.12}$$

In this work, extending the signal by means of simple repetition has proven as the most promising approach. The cyclic autocorrelation measure, being defined by

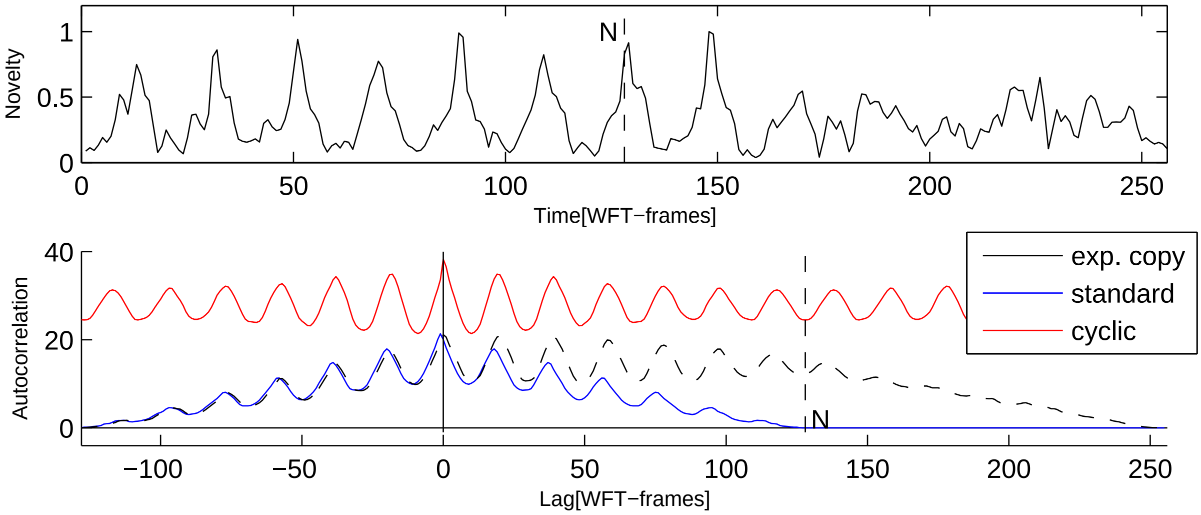


Figure 4.9: Top: excerpt of above $F_{\mathrm{NOV}\uparrow}$ novelty measure. Bottom: comparison of different autocorrelation measures, top-down: $2N$-cyclic autocorrelation (in red). Correlation of signal $x$ and double length excerpt $x'$ (in dashed black). Autocorrelation using a zero padded copy $x'$ (blue).

$$\begin{aligned}\operatorname{acorr}^{\circ}_x(\tau) &= \sum_{k=0}^{N-1} x(k)x((k+\tau) \bmod N) \\ &\Leftrightarrow (x(\cdot) *_N x(-\cdot))(\tau),\end{aligned} \tag{4.13}$$

for $x \in \mathbb{R}^N$ and $\tau \in \{0, \cdots, N-1\}$, is used. This procedure leads to major improvements in temporal resolution and computational costs: first, only signal values covered by the actual window position are used. This is particularly useful for the following reason: although some animal sounds contain very precisely timed periodic elements, the possibility for small deviations, leading to blurred autocorrelation curves, increases with observation time. Consequently, shortening the autocorrelation window results in sharper features. Moreover, the usage of the fast Fourier transform leads to a significant speedup in convolution time. Because of the signal padding mentioned above, and the use of the convolution workaround described in Chapter 3.3.1, an FFT of length $4N$ is necessary to compute the expanded signal's autocorrelation (4.12). Exploiting the fact that DFT-based convolution is naturally cyclic, (4.13) is calculated using a $\mathrm{FFT}_N$. As the cyclic autocorrelation of $x \in \mathbb{R}^N$, due to the periodicity of cyclic convolution, is $N$-periodic and, as an effect of the convolution with itself, symmetric at $\tau = \frac{N}{2}$, only $\frac{N}{2}$ of the coefficients contain valuable information. In order to acquire autocorrelation values for lags $0 \leq \tau < N$, we autocorrelate signal excerpts of length $2N$, as depicted by the top red curve in Figure 4.9.

The actual length $s_{ac}$ of the autocorrelated vector is determined by the framerate fps and the maximal period to be detected. As explained above, using a window size of $s_{ac} = 2\tau + 1$ yields autocorrelation coefficients for lags up to the desired period. An autocorrelation window twice this period's length is bound to contain up to two of such periodic elements. These usually only provide a weak basis for the measurement of periodics. Thus there usually is a tradeoff between a long autocorrelation window, leading to more robust autocorrelation measures, and the decrease of the curve's sharpness. In this thesis, a frame rate of fps = 300 WFT-frames per second and an autocorrelation length of 64 to 128 WFT-frames is used, to capture the 50 Hz periodics of Savi's Warbler, although the autocorrelation using a window length of $s_{ac}^{\min} = 2 \cdot \frac{\text{fps}}{50} = 12$ would still contain the lags associated to the desired period. In our other subsequently introduced detectors, a FFT-length of $N = 256$ samples is used for the convolution, revealing autocorrelation coefficients for lags between $-128 \leq \tau < 128$ samples. In order to discard the symmetric components, the features only contain information on positive lags for $\tau \in \{0, \cdots, \frac{N}{2} - 1\}$. Assuming a stepwidth $\Delta_{ac}$, the autocorrelation sequence for novelty curves $x \in \ell^2(\mathbb{Z})$ is defined as

$$F_{\text{ACORR}}(\tau, k) = \text{acorr}^{\circ}{}_{x_k}(\tau), \quad \text{where} \qquad (4.14)$$
$$x_k := (x(t_k), x(t_k + 1), \cdots, x(t_k + s_{ac} - 1))^{\top}$$

denotes the signal excerpt after pointwise multiplication with a rectangular window, which is aligned at position $t_k := t_s + k \cdot \Delta_{ac}$. For each individual ACORR-frame, the autocorrelation curve is then normalised to a maximum value of 1. This is done for several reasons: As the maximum autocorrelation value, corresponding to the signal's energy, is contained in the zero-lag autocorrelation, the $F_{\text{ACORR}}(\cdot, k)$ curves are weighted by the squared novelty sum of frame $x_k$,

$$F_{\text{ACORR}}^{\text{norm}}(\tau, k) = \frac{F_{\text{ACORR}}(\tau, k)}{F_{\text{ACORR}}(0, k)}. \qquad (4.15)$$

This normalisation will preserve the information about periodic elements occurring in the analysis window $x_k$, while dismissing the overall signal's strength. Thus, the normalised autocorrelation features are quite invariant up to the overall signals strength. Although the depicted autocorrelation curves will further on depict the zero-lag autocorrelation to have major energy, the reader is suggested to focus the attention on the remaining local autocorrelation maxima. Such normalisation furthermore is a simple measure to make the $F_{\text{ACORR}}$ data accessible to further learnable classificators using Gaussian Mixture Models (GMM's) as discussed in Chapter 5.4.2.

In Figure 4.10, the normalised autocorrelation sequence of a Chaffinch stanza is depicted. Following the first autocorrelation maxima through the whole sequence, one can easily see that the position or index of the respective maxima also approximates the temporal distance of the birdsong elements depicted by the above spectral features. Furthermore, the resulting subsequences of very similar autocorrelation vectors clearly reflect the phrase structure, being sung by the targeted bird. Surely, this nature of the autocorrelation sequence will be exploited in the structure extraction Chapter (5.4.1). The $F_{\text{ACORR}}$ and $F_{\text{ACORR}}^{\text{norm}}$ features are quite

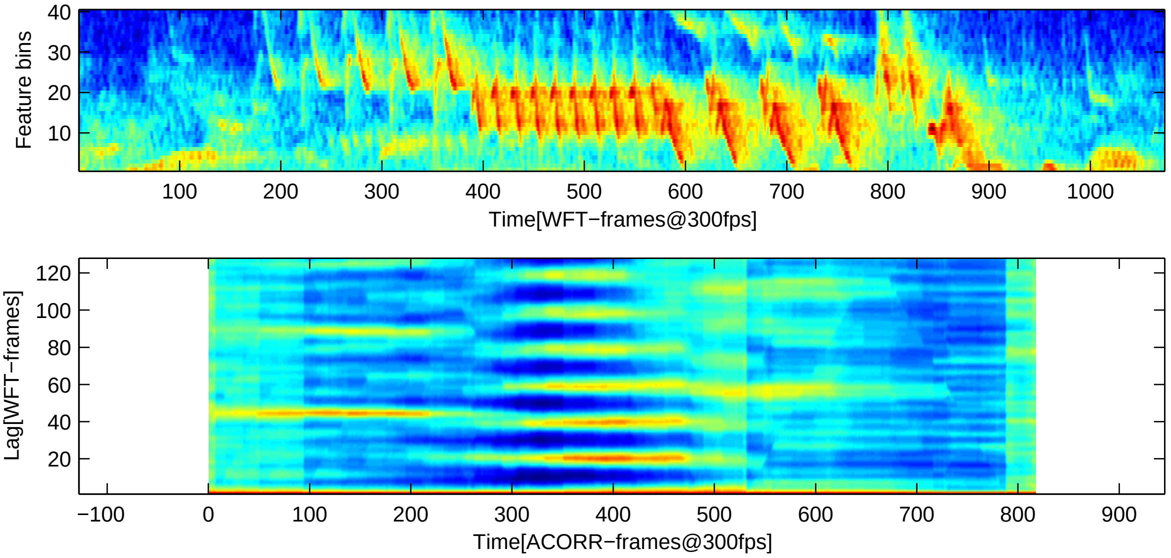


Figure 4.10: Top: spectral features of the Chaffinch stanza example. Bottom: $F_{\text{ACORR}}^{\text{norm}}$ features derived from these features. Although the features are derived with a stepwidth of a single frame ($\text{fps}_{\text{ac}} = 300$), the window size of ($N = 256$) samples used for the $N$-cyclic autocorrelation causes the reduced amount of data.

robust concerning noisy novelty curves, because peaks associated with signal noise seldom appear at constant time intervals. Thus they do not accumulate during the calculation of $F_{\text{ACORR}}(\tau)$ values, as a periodic signal would do. Countermeasures concerning periodic noise will be discussed in Chapter 4.2.7. Another useful property of the autocorrelation measure is that, besides indicating a fundamental period of a periodic sequence, multiples $n\tau, n \in \mathbb{N}$ of this estimated period are likely to have high autocorrelation values. In such a case, the assumption of an analysed sequence containing more than two uniformly spaced elements is substantiated. Thus, this sequence is to be considered to contain a periodic signal. The number of those additional local maxima is bounded by the length of the autocorrelated signal and the maximum multiple of the fundamental period fitting into the ACORR-frame. Given an appropriate parametrisation for the $F_{\text{ACORR}}$ features, a sequence containing $c$ element repetitions usually produces $F_{\text{ACORR}}$-peaks at positions $\{n\tau \mid 1 \leq n < c\}$.

### 4.2.3 Autocorrelation sharpness

Although containing robust information about periodic elements, the $F_{\text{ACORR}}$ and $F_{\text{ACORR}}^{\text{norm}}$ features fail to deliver any information concerning the signal's volume or signal-to-noise ratio. Unfortunately, this leads to false period detections in quiet, noise-dominated signals. As this kind of signal is frequently recorded during late night monitoring sessions, one must find another useful feature, providing helpful information concerning the mentioned properties, which can be combined with the above autocorrelation features. The considered measure should also be invariant up to the absolute signal level, as, sometimes, the signal gain changes during the monitoring recording. An useful indicator, measuring the ratio between periodic signals and non-periodic noise, is the *Autocorrelation sharpness*. This feature is derived from the *Spectral flatness* measure (4.2), which is an indicator for a homogeneous distribution of energy in the spectrum. Applying the specflat measure on autocorrelation curves, we need to calculate the mean autocorrelation coefficient

$$\overline{F_{\mathrm{ACORR}}}(k) := \frac{\sum_{\tau=0}^{s_{ac}-1} F_{\mathrm{ACORR}}(\tau, k)}{s_{ac}}$$

and the geometric mean:

$$\breve{F}_{\mathrm{ACORR}}(k) := \sqrt[s_{ac}]{\prod_{\tau=0}^{s_{ac}-1} F_{\mathrm{ACORR}}(\tau, k)}$$

Thus, autocorrelation sharpness is defined as

$$F_{\mathrm{ACSHARP}}(k) := 1 - \frac{\breve{F}_{\mathrm{ACORR}}(k)}{\overline{F_{\mathrm{ACORR}}}(k)}. \tag{4.16}$$

It is easily proven that the above measure for all $k$ is bounded by $0 \leq F_{\mathrm{ACSHARP}}(k) \leq 1$. The normalisation used in (4.15) does not affect the above measure. Hence, the $F_{\mathrm{ACSHARP}}$-measure may be derived from the normalised autocorrelation features, alternatively. A high autocorrelation sharpness value indicates the signal's energy being concentrated in a single sequence of periodic elements.

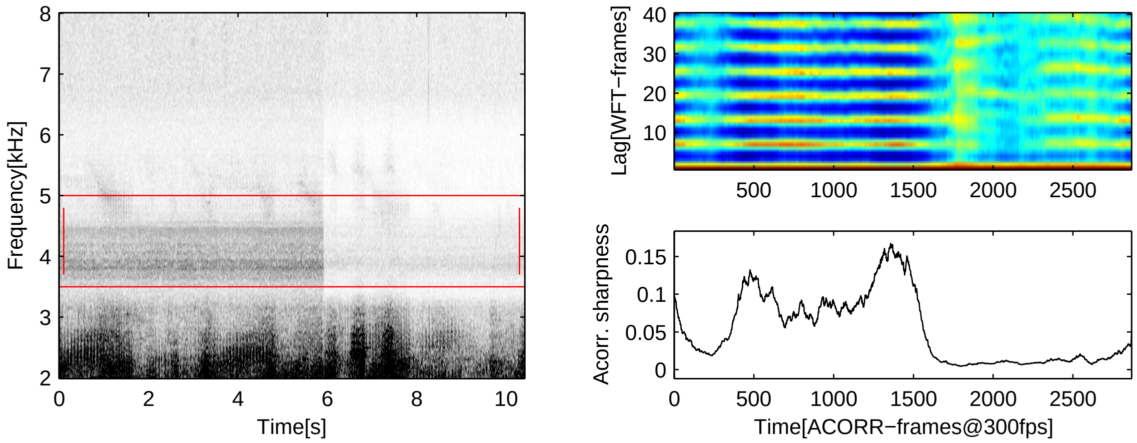


Figure 4.11: Demonstration of the $F_{\mathrm{ACSHARP}}$ feature: Left: spectrogram containing six seconds of a strong Savi's Warbler's song. Top right: autocorrelation sequence of the frequency band (3.5-5 kHz). Bottom right: associated autocorrelation sharpness measure. In the spectrogram, the very fast (50 Hz) element repetition of the bird's song is blurred.

### 4.2.4 Abp-features

Although, for some applications, calculating a single novelty curve and autocorrelation series may be sufficient to obtain an description of the occurring periodic elements, our experiments have shown that this usually is not the case with undirected monitoring recordings. These, naturally containing lots of coexisting sounds and noise, usually produce suboptimal novelty and autocorrelation results. In order to get a series of more sharp autocorrelation curves, we apply a subband autocorrelation strategy: at first, the analysed band $B_l$ is splitted into subbands $S(B_l)_m$, for $m \in \{0, 1, \cdots, \#_{\text{sbands}} - 1\}$. The frequency ranges of these bands are designed to overlap each other about a half of their bandwidth. For $\#_{\text{sbands}} = 5$ or 9, and $\#_{\text{bins}} = 30$, each subband is associated to $\Delta_{sb} = 10$ and 6 spectral feature bins, respectively, corresponding to the following coefficients:

$$S(B_l)_m := \{j + \lfloor m \cdot s_{sb} \rfloor \mid 0 \leq j < \Delta_{sb}\}. \tag{4.17}$$

Here, the stepwidth is set to $s_{sb} = 3$ for 5 subbands, and $s_{sb} = 5$ for a division into 9 bands. In a next step, the subband novelty curves are derived, and a $F^{\text{norm}}_{\text{ACORR}}$ sequence is computed for each subband. The final autocorrelation sequence is now composed of these sequences in the following manner: for each ACORR-frame $k$, we choose one of the $\#_{\text{bins}}$ alternative autocorrelation curves. To this end, we use the *Autocorrelation sharpness* measure $F_{\text{ACSHARP}}$ to define a quality ranking of these curves on a frame-wise basis. In prior to the selection, the resulting autocorrelation sharpness curves are smoothed by means of convolution with a Hann window, as defined in (3.27). For each frame, the sharpest autocorrelation curve is then selected and saved into the mixed autocorrelation sequence. For the case of three subbands, this is depicted in Figure 4.12. Let $F^{\text{norm}}_{\text{ACORR}}[m](\tau, \cdot)$ and $F_{\text{ACSHARP}}[m](\cdot)$ represent the normalised autocorrelation sequence and autocorrelation sharpness curve of subband $S(B_l)_m$. The adaptive band periodicity features $F_{\text{ABP}}$ are then formally defined as

$$F_{\text{ABP}}(\tau, k) := F^{\text{norm}}_{\text{ACORR}}[M_k](\tau, k), \tag{4.18}$$

$$\text{with} \quad M_k := \arg\max_m \left(F_{\text{ACSHARP}}[m](k)\right). \tag{4.19}$$

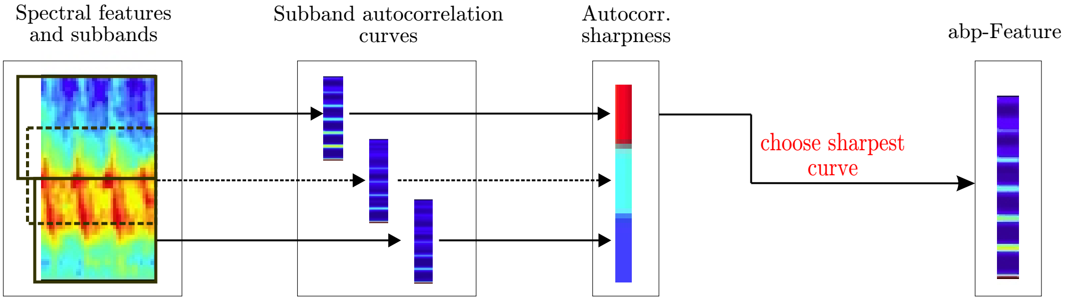


Figure 4.12: Extraction of an $F_{\text{ABP}}$ frame. Here the autocorrelation of the topmost subband has a superior autocorrelation sharpness value.

As the subband indices $M_k$ (see top of Figure 4.13) solely depend on the $F_{\text{ACSHARP}}[m](k)$ features of a single frame, the resulting index sequence may contain frequent index changes caused by noise or single bad $F_{\text{ACORR}}$ curves. The resulting $F_{\text{ABP}}$-sequence is likely to contain many unintended discontinuities between adjacent signal frames, which should be avoided. Such distortions are minimised by the preliminary smoothing of the $F_{\text{ACSHARP}}[m](\cdot)$ feature sequence. There are also examples for intended band index changes. E.g., the typical Chaffinch's stanza contains several phrases, consisting of elements with different spectral bandwidths. In the optimum case, the subband index sequence $(M_k)$ should remain stable for inner-phrase frames whereas the index changes are located during frames containing a phrase-transition. The Hann window's width should be chosen to be compatible with the typical phrase structure of the song to be detected. Otherwise, an oversized window will result in delayed or even skipped phrase transitions. This would contradict the adaptive design of the desired features.

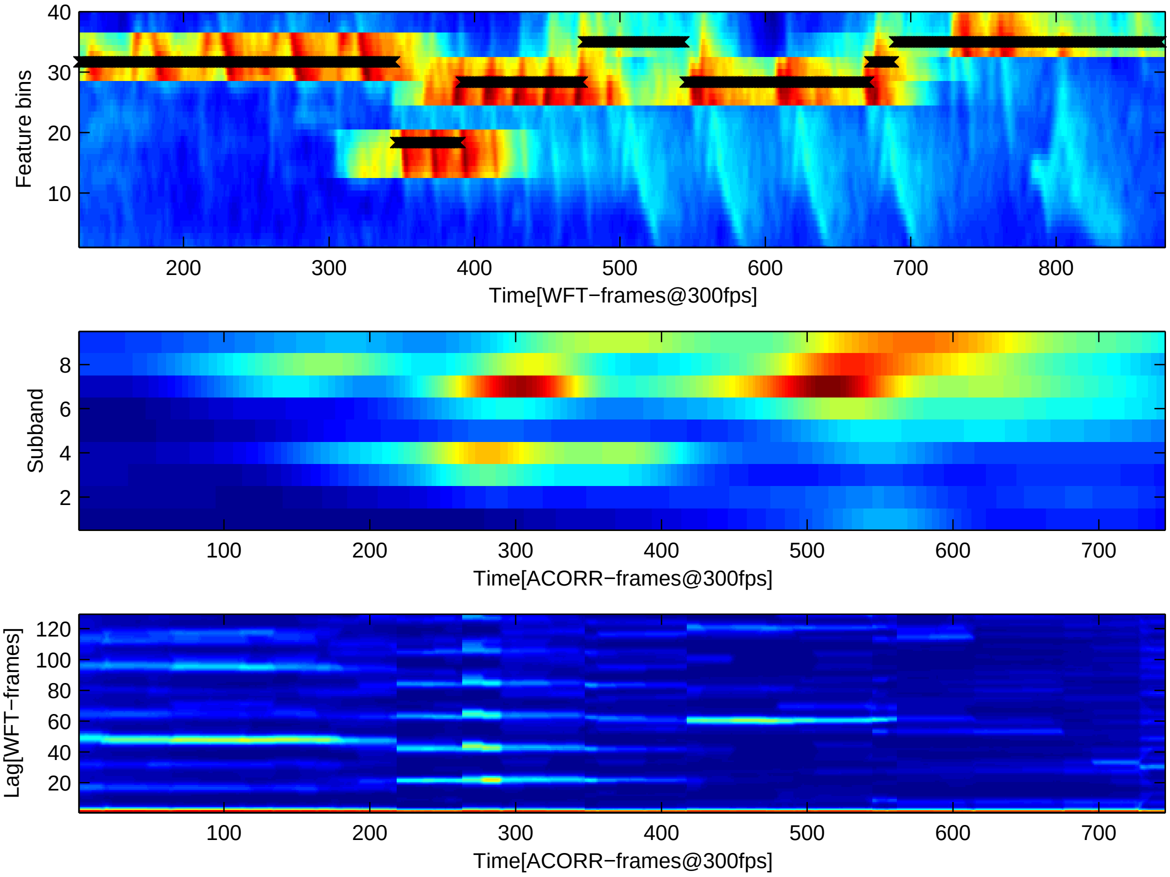


Figure 4.13: Example continued from figure 4.8 (p.49). Top: spectral features. For each frame, the subband index $M_k$, corresponding to the highest $F_{\mathrm{ACSHARP}}$ measure, is indicated through a black cross (see Equation 4.19). The highlighted areas correspond to the data contributing to the autocorrelation windows. Middle: $F_{\mathrm{ACSHARP}}$ measures for all subbands, blue colours correspond to low, red refers to high values. Bottom: Resulting $F_{\mathrm{ABP}}$ autocorrelation sequence.

Thus, as depicted in Figure 4.13, we obtain a sequence of the sharpest autocorrelation curves, expressing the pitches contained in the respective subband. A comparison of both the common and enhanced autocorrelation sequence can be found in Figure 4.14.As these curves are substituted for the $F_{\mathrm{ACORR}}$ features, representing a whole band's periodics, one has to choose that band's borders quite carefully. If an oversized bandwidth is chosen for $B_l$, the possibility of expressing the periodics of unintended signals is significantly higher. Usually, this band is of adequate size if it at least partially covers the frequency range of all expected elements. Full containment is not required. Assumed that most of the expected bird sounds have a band-

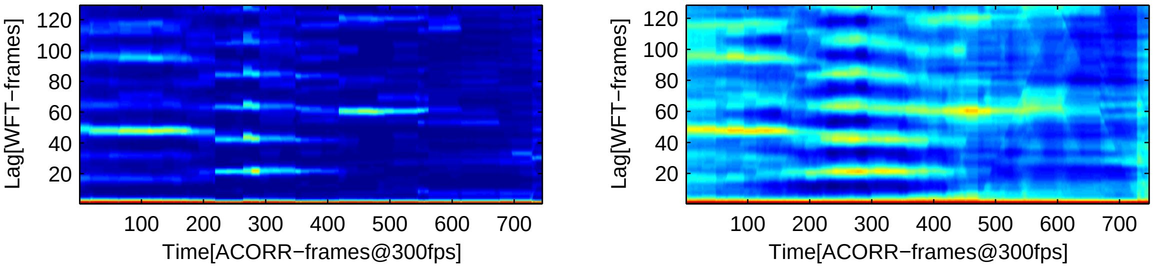


Figure 4.14: Comparison of autocorrelation sequences. Left: adaptive band periodic ($F_{\text{ABP}}$) features. Right: $F_{\text{ACORR}}$ autocorrelation sequence.

width at least of the chosen band's size, the selected autocorrelation curves of the subbands will contain a sharper representation of the periods while including the information about the periodic elements we are looking for. With focus on the processing time, the procedure consumes at least $\#_{\text{sbands}}$ times the CPU time needed to extract an ordinary $F_{\text{ACORR}}$ feature sequence for calculating the subband autocorrelation curves and $F_{\text{ACSHARP}}$ features.

### 4.2.5 Reducing the dimension of autocorrelation data

In order to make the information contained in the autocorrelation features accessible to unsupervised learning algorithms, the original dimensionality of the $F^{\text{norm}}_{\text{ACORR}}$ features may have to be reduced. This is particularly the case for the Expectation-Maximisation (EM) strategy used to estimate a hidden Markov model in section 5.4.2. The methods used for this reduction are outlined in the following.

As a first step, the autocorrelation curves are shortened to contain the minimal range of lags being sufficient to contain the desired periodics. Now, two different approaches can be used to achieve a further reduction of dimensions: downsampling the autocorrelation curves comes with a coarsening of the distinct repetition periods. On the one hand, this helps the generalisation needed to build a robust model, whilst, on the other hand, excessively coarsening the lag scale may eliminate most of the information inherent to these autocorrelation curves. Furthermore, reducing the sample rate of these curves implies a reduction of the highest representable frequency. In the experiments performed for this work, autocorrelation curves are computed from spectral features with high temporal resolution (fps = 150). Now, the maximum lag is defined as 59 WFT-frames, corresponding to an element repetition rate of 2.6 Hz. Downsampling the autocorrelation curves by a factor of 5 leads to a coarse representation contained in 12 coefficients, roughly capturing periodics from 2.6 to 15 Hz. Figure 4.15 displays this change in lag resolution. Being derived from the previous example, the sequence's temporal resolution still amounts to 300 frames per second.

Another popular dimension reduction technique is given by the *Principal Component Analysis* (PCA). The PCA of a data set defines an orthogonal transform providing a decomposition of the data, where the variance, and thus the information enclosed in the first $n$ components is always maximal. A popular technique is then to discard some of the last coefficients of the transformed data, carrying small information on the whole data set. However, the new base vectors, and thus the transformation matrix, depend on the initial training data set. Thus, vectors not contained in the training data are likely to be badly represented by the new

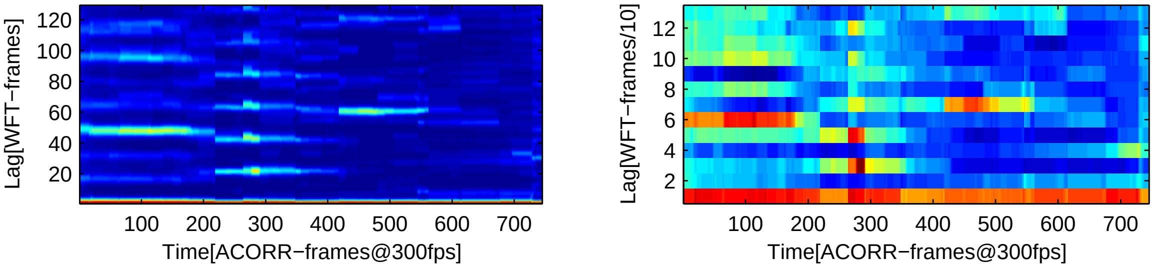


Figure 4.15: Dimension reduction of autocorrelation sequences. Left: adaptive band periodic ($F_{\mathrm{ABP}}$) features. Right: 10 times downsampled $F_{\mathrm{ABP}}$ features.

base. In case of the training data containing autocorrelation sequences from one or several Chaffinch songs, just a subset of all relevant autocorrelation curves is contained. Although our experiments showed a reasonable representation of the initial data set, even when only using the 5 first coefficients of the transformed data, stanzas of new individuals were represented very poorly, due to the slightly differing periodics.

Thus, for the experiments described in this work, being aimed at a general model covering the songs of multiple unknown individuals, the resampling approach seems more promising. However, for applications concerned with the representation of the stanzas of a single individual, the PCA might perform quite well by reducing the redundancy inherent in the plain autocorrelation curves.

### 4.2.6 Fourier transformed autocorrelation curves

On the one hand, as described in Section 4.2.2, the autocorrelation sequence already constitutes an intuitive representation of periodic elements.On the other hand, however, it also contains some redundancy, caused by $\tau$-periodic elements having multiple repetitions, though producing repeated autocorrelation maxima.By transforming the autocorrelation curves into the frequency domain, this redundancy is utilised to get a robust representation of the periodic elements' frequencies. As the additional peaks are located at equidistant positions $\{n\tau \mid 1 \leq n < c\}$, the subsequent fourier transform reproduces this periodicity as a single peak, e.g. attributed to the cosine curve $\cos(2\pi\frac{t}{n})$. In comparison to the autocorrelation curves, the envisaged kind of representation (see Fig. 4.16) features a greater accessibility to an intuitive comprehension. Prior to the fourier transformation, in order to minimize aliasing effects being caused by high autocorrelation values at the ultimate lags, the curves are pointwise multiplied with a Hann window of the same length, as described in Section 3.3.3. Now, a fast Fourier transform (e.g. of length $s_{ps} = N = 256$) is performed. To fit the FFT size, the windowed $F_{\mathrm{ACORR}}$- or $F_{\mathrm{ABP}}$-features are padded with zeros.

$$F_{\mathrm{NPS}}(\cdot,k) := \mathrm{DFT}_N\,\widetilde{F_{\mathrm{ABP}}}(\cdot,k), \tag{4.20}$$

$$\text{where}\quad \widetilde{F_{\mathrm{ABP}}}(\tau,k) := \begin{cases} g(\tau)\cdot F_{\mathrm{ABP}}(\tau,k) & 0 \leq \tau < s_{ac}, \\ 0 & s_{ac} \leq \tau < N. \end{cases}$$

Here, $N > s_{ac}$ is assumed. $g(\tau)$ refers to the Hann window defined in Equation (3.27) and $s_{ac}$ denotes the length of an autocorrelation vector. For the general autocorrelation measure defined in (4.13), this procedure leads back to the power spectrum or absolute value of the Fourier transformed novelty curves:

$$\begin{aligned}\mathrm{DFT}_N \,\mathrm{acorr}^{\circ}_{x_k}(\cdot) &= \mathrm{DFT}_N \,\mathrm{IDFT}_N X_k \cdot \overline{X_k} \\ &= X_k \cdot \overline{X_k} \\ &= \|X_k\|^2.\end{aligned}$$

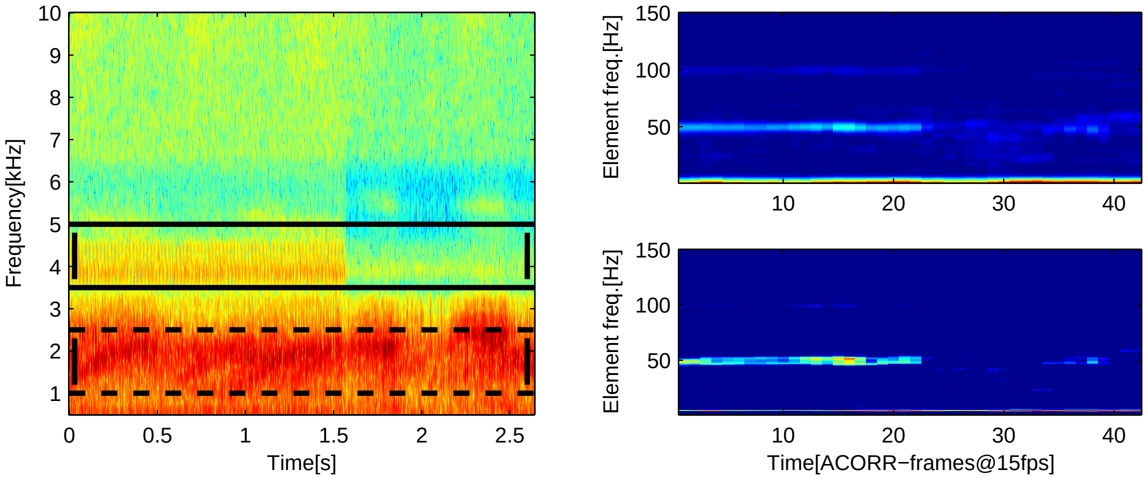


Figure 4.16: Novelty power spectrum features comparison. Left: spectrogram containing Savi's Warbler's song. Boxes specify the feature band (upper band) and the flanking band (dashed). Top right: $F_{\mathrm{NPS}}$ novelty power spectrum features. Bottom right: denoised $F^{-\mathrm{nois}}_{\mathrm{NPS}}$ features, using the flanking band.

Note that $X_k$ represents the DFT-transformed excerpt $x_k$ of a novelty curve. In the following, the $F_{\mathrm{NPS}}$ features will be referenced as novelty power spectrum (NPS) features. A major advantage of the features proposed in (4.20) is the easy way they can be interpreted: a peak at position $F_{\mathrm{NPS}}(f,k)$ indicates a likely periodicity with an element repetition frequency of $\frac{f}{s_{ps}} \cdot \mathrm{fps}$ Hz. Unlike the autocorrelation features, there are no subharmonics (e.g. at positions $\{\frac{f}{2^n} \mid n \in \mathbb{N}\}$) contained in the $F_{\mathrm{NPS}}$ features. The first fourier coefficient contains the mean autocorrelation energy, which will be of no further use in this thesis.

### 4.2.7 Cancellation of periodic noise in $F_{\mathrm{NPS}}$ features

In many applications, knowledge about the expected frequency range of a particular animal is an important prerequisite to the identification task. In this section, utilising such knowledge, information about periodic background noise is gathered. A *flanking* frequency band near, but surely disjunct to the birds typical frequency range is analysed for periodic elements, and $F_{\mathrm{NPS}}$ features are extracted. Now, broadband background signals present in both the flanking band and the birds band are removed from the latter band's features by means of subtraction. Let $F_{\mathrm{NPS}}[b]$ and $F_{\mathrm{NPS}}[f]$ contain the power spectrum features of the bird's and the flanking band signals. The denoised $F^{-\mathrm{nois}}_{\mathrm{NPS}}$ features are then defined as

$$F_{\mathrm{NPS}}^{-\mathrm{nois}}(j,k) = \min\left(F_{\mathrm{NPS}}[b](j,k) - F_{\mathrm{NPS}}[f](j,k),\, 0\right). \tag{4.21}$$

In Figure 4.16, a monitoring signal containing a Savi's warblers song and several calling tree frogs is analysed using the features introduced above. In the top right image, the novelty power spectrum features are plotted against an element repetition frequency. As the underlying spectral features are derived using a frame rate of fps = 300, the ultimate detectable frequency determining the repetition of the bird's song's elements is bound to 150 Hz, corresponding to an autocorrelation maximum at the lag of $\tau = 2$. Note that this frequency is not to be mistaken as the spectrogram's frequency. The element repetition frequency indicates a probability for the repetitive occurrence of acoustical events being visible in the spectrogram. In this example, the ACORR-frame rate of the Savi's Warbler detector is used, saving computation time by reducing the framerate to $\mathrm{fps}_{\mathrm{ac}} = 15$ autocorrelation curves, resulting in the same amount of $F_{\mathrm{NPS}}$ features per second. The bottom right image displays the denoised power spectrum features, using the frequency band between 1 and 2.5 kHz as flanking band. Comparing the latter image with the former $F_{\mathrm{NPS}}$ features, the signal to noise ratio has clearly been improved. Usually, situating the flanking band below the bird's band is quite secure, because, as our experiments have shown, harmonics, evolving from the bird itself or machine-induced distortion, as well as other correlated song components, are likely to distort the upper frequency bands. Especially when working with high noise levels, the detection of such a complementary band is a challenging task.

Before discussing the actual detectors, the reader may reconsider the feature dependency chart which is also displayed the beginning of this chapter.

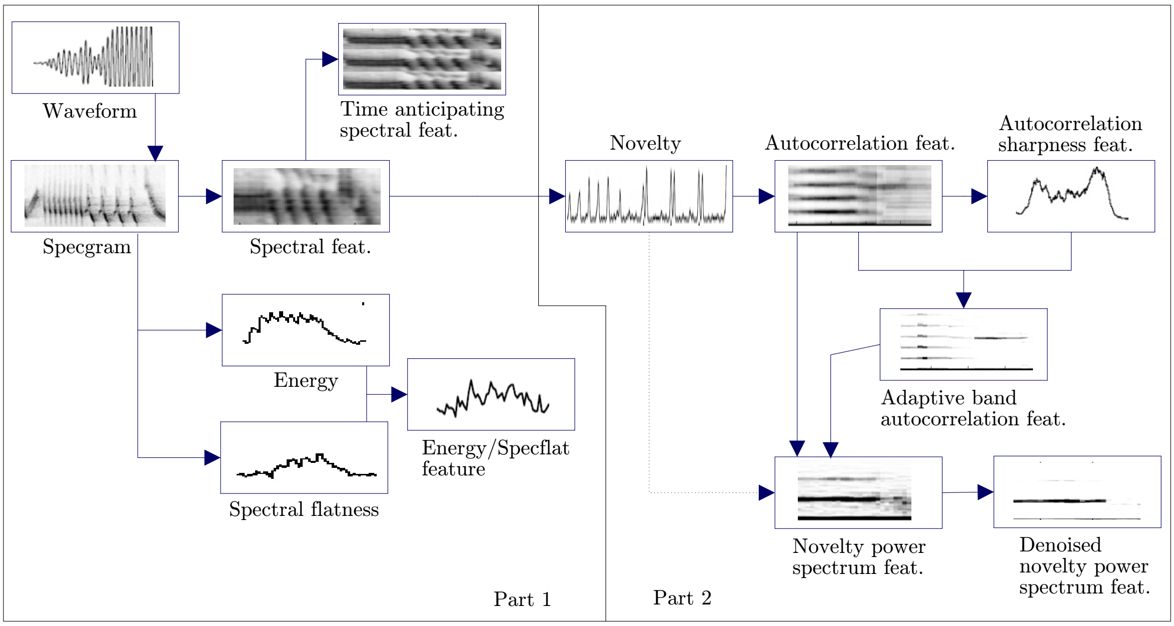

Waveform
Time anticipating spectral feat.
Specgram
Spectral feat.
Energy
Energy/Specflat feature
Spectral flatness
Part 1
Novelty
Autocorrelation feat.
Autocorrelation sharpness feat.
Adaptive band autocorrelation feat.
Novelty power spectrum feat.
Denoised novelty power spectrum feat.
Part 2

Figure 4.17: Reminder: chart of feature dependencies from Chapter 4.

# Chapter 5

# Signal classification and birdsong recognition

In this chapter, the features described in the previous chapters will be used to solve several classification tasks. Considering bird song recognition, there are several modes of classification to be discussed. Building a binary classifier for the decision whether a selected signal segment contains a song of a specific species or not seems to be a good start. In this case, the audio data needed for evaluating and eventually training the classifier is not required to contain less recordings of other bird species. In the case of bird recordings, this is a clear advantage, because there are few annotated monitoring recordings available and manual annotation turns out to be quite time-consuming and therefore too expensive.

The two main detectors to be discussed in this chapter use an extensive as well as diverse set of signal processing and pattern recognition techniques. Fortunately, most of these techniques can be grouped at distinct levels of abstraction. In order to obtain a comprehensible structure, the following sections are ordered by means of paradigms. In fact, both of the detectors are explained in two steps. The descriptions of the preselection routines are combined in Section 5.2, which discusses a "prefiltering" paradigm. Here, the topics of interpreting one-dimensional features measuring energy or spectral flatness, as well as the matching of spectral feature sequences using Dynamic Time Warping, are covered. Thereafter, Section 5.3 approaches the main detection routine of the Savi's Warbler detector. Section 5.4 motivates the use of advanced birdsong models through the detailed description of the Chaffinch detector's classification routine. Providing the essential superstructure of the named detectors, the next section follows with motivational character.

## 5.1 Overview for Chaffinch and Savi's Warbler detectors

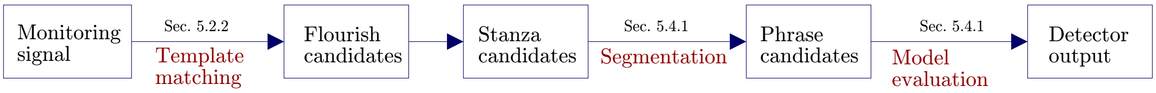


Figure 5.1: Overview of the Chaffinch detector.

**Detecting the Chaffinch** As described in Chapter 2.4, the Chaffinch's song has a characteristic flourish at the very end of almost each stanza. Motivated by this fact, the first step in the Chaffinch detector is to detect elements which are possibly representing such a flourish. This is achieved by searching for a set of spectral feature templates, representing a couple of typical flourishes. The searching procedure uses a DTW-based distance measure to identify possible flourish candidates in the record to be analysed. Once extracted, the latter segments can be grouped into clusters of similar flourishes.

Now, the $F_{\text{ABP}}$ autocorrelation features are used to extract an element repetition period for each ACORR-frame. Being computationally expensive, this procedure is only applied on stanza candidates attached to the above candidate flourish segments. Now, separately processing each stanza candidate, the knowledge gained from the Chaffinch stanza segmentation study (Chap. 2.4.2) is used to implement a Chaffinch song model: at first, a segmentation is performed on the sequence of repetition periods, extracting segments with parameters similar to Chaffinch phrases. In a second step, several conditions are checked concerning the temporal localisation of the extracted segments. The application of both of the previous steps can be interpreted as the comparison of the candidate to a Chaffinch stanza model, imposing several conditions on the parameters of the contained phrases (see Figure 2.7, p.20 for terminology). In case that all conditions are fulfilled, the candidate is classified as Chaffinch and given a ranking value.

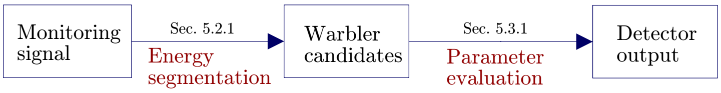


Figure 5.2: Overview of the Savi's Warbler detector.

**Savi's Warbler detector** Similar to the above procedure, a preselection routine is performed prior to the expensive classification. For the Savi's Warbler detector, the time-varying energy curve of the bird's typical frequency band is extracted. Using a special segmentation routine, signal parts of major energy are extracted until a sufficient amount of excerpts is gathered. This procedure reflects the fact of the targeted bird singing for long periods. Here, though only analysing the most promising excerpts, the detection of a singing individual is still likely. When evaluating the overall stanza duration of these birds, the preselection step may be omitted at the expense of computation time.

The main classification routine of the discussed detector uses an adaptation of the denoised novelty power spectrum ($F_{\text{NPS}}^{-\text{nois}}$) features, providing 5 parallel streams of denoised features. Then, in particular two essential features, measuring the element repetition period as well as the presence of the allegedly periodic signal, are extracted on an ACORR-frame basis. Frames not fitting the expected values for these features are discarded from further processing. Finally, a cascaded segmentation is performed on the residual frames, delivering segments to be classified as parts of Savi's Warber songs. Also, a ranking value is attached to each of these segments.

## 5.2 Directing the search via segment preselection

As some of the detectors presented in the subsequent sections are quite expensive from a computational point of view, we introduce a two stage approach for these detectors: in the first stage, *interesting segments* are extracted. These segments are selected by a computationally cheap classifier, whose decisions are based on low-level signal measurements. Afterwards, the second classifier is used to decide on whether the preselected segments contain the sought-after bird songs or not. Usually, signal excerpts only reference a small part of the whole signal. Thus, the second and most expensive part of the computation is performed on a reduced data set. Depending on the actual signal source and low-level measurements, this approach saves a lot of time. As the first classifier is not expected to perform an accurate detection of a specific bird species, its false positive rate is allowed to be very high. On the contrary, the number of stanzas being falsely sorted out at this stage has to be minimized. Otherwise, many segments will remain false negatives without a possible and costly third (re-) classification stage. Such a three-stage classification approach, where an adapted classifier is used to re-classify the whole signal, will not be discussed in this work.

Because, as described above, the false negative rate of the first classificator is critical for an accurate overall performance, the two-stage approach should not be used where quantitative values, such as the detection rate per hour, are of interest. The Savi's Warbler detector was initially designed to extract reference segments, to be used as proof of presence of a Warbler's song, from recordings usually longer than one hour. These segments are used as proof for the presence of the bird in the surrounding area of the recording microphone. In this scenario, the proposed preliminary extraction of candidate segments is quite useful.

### 5.2.1 Energy and spectral flatness features

**Savi's Warblers Energy classificator** The band energy feature described in equation (4.1) is a cheaply computable feature. In the Savi's Warbler detector, an energy measure is used as an indicator for acoustic events within the Warbler's typical frequency band. As the Warbler's song is likely to raise the bands signal energy, we will extract areas of high energy as interesting segments. Thus, the first step in the respective detector is to use an adaptive threshold criterion for the extraction of such segments. The segmentation is performed as follows.

At first, an energy curve of the frequency band in the range of 3,5 to 5 kHz is calculated. Now, for reasons of memory management, this curve is downsampled to a sampling frequency of 2 Hz. In order to minimise the aliasing effects associated with this procedure, the energy curve is low-pass filtered in advance. The resampled signal is then cleaned from local discontinuities by means of applying a median filter. The filter has a length of 3 samples.

$$\mathrm{medfilt}_x(n) := \begin{cases} \mathrm{median}\{x(n-1), x(n), (n+1)\}, & \text{for } 1 \leq n < N-1, \\ x(n), & \text{otherwise.} \end{cases} \tag{5.1}$$

The following segmentation algorithm is designed to extract at least *an eighth* of the input signal's length as interesting segments. These are selected to contain relatively high energy measurements. Thereby, the length of these segments and the steadiness of the energy curve inside the segment boundaries are maximized. This procedure is motivated by the fact of Savi's Warbler's stanzas being quite steady and long. The algorithm described below is designed to be used with signals which are longer than one hour. As the Savi's Warbler is likely to sing for a long time, easily exceeding 20 minutes, the minimum excerpt length mentioned above is sufficient for a general detection of the bird's song. In a first step, we try to extract rather long and very steady segments from the signal. If the cumulative length of the extracted segments does not suffice the length criterion mentioned above, the thresholds determining the minimum segment size and signal steadiness are lowered. Again, the yet unsegmented parts of the signal are processed in the same manner as described above. Thus, the set of candidate segments is enlarged during each iteration. This procedure is repeated until either a sufficient cumulative segment length is achieved or, regarding the segment size, a minimal length threshold is reached.

For the description of the actual algorithm, we need to define *segments* and sets of segments: a segment $u$ constitutes a pair, containing the sample positions of its first and last constituting time index: $u := (f, l)$, $f, l \in \mathbb{Z}$. As the segmentation is based on the energy curve, the 2 Hz sampled frames will serve as the unit of reference. Now, the segment set $S$ will be used to gather the detected interesting segments, while the set $U$ will keep the unsegmented rest. The steadiness of an energy vector $x \in \mathbb{C}^N$ mentioned above is approximated using an averaging filter of variable length $s$ as defined in Equation (3.23). The filtered signal expresses the mean signal energy measured in a fixed time window. Now, for $n \in \{0, \cdots, N-1\}$, we define the binary decision vector

$$\mathrm{goodpos}_x^s(n) := \begin{cases} 1 & \text{if } \mathrm{medfilt}_x(n) > 0.98\, \overset{s}{\overline{\mathrm{medfilt}_x}}\,(n), \\ 0 & \text{otherwise}, \end{cases} \tag{5.2}$$

indicating an energy sample exceeding a fixed steadiness threshold. When analysing a multichannel recording $x[c](n)$, e.g. for $c \in \{0, \cdots, 3\}$, the filtering is applied on each of the individual channels, delivering the decision vectors $\mathrm{goodpos}_{x[c]}^s(n)$. Afterwards, the results of the above criterion are combined by means of logical disjunction:

$$\mathrm{goodpos}_x^s(n) = \bigvee_c \mathrm{goodpos}_{x[c]}^s(n).$$

The actual implementation follows Algorithm 1. In line 17 of this algorithm, the operator $\setminus^*$ is used in a non-standard way, as shown in Figure 5.4: actually, in $U = U \setminus^* (p, q)$, the segment $(f_i, l_i)$ containing the time positions $(p, q)$ is identified. Then, this segment is split into two new segments $(f_i, p-1)$ and $(q+1, l_i)$. The initial segment $(f_i, l_i) \in U$ is now replaced the two new segments. If one of these segments is to short, it is omitted. The resulting segmentation is depicted in Figure 5.3, showing a 4-channel recording's energy curves. In the second classification stage of the Savi's Warbler detector described in Section 5.3, only the segments contained in the set $S$ are processed. In the case of $S$ remaining empty, the first 10 minutes of the monitoring signal are classified.

**Spectral flatness indicator.** The band energy segmentation algorithm can also be applied on a $F_{\mathrm{ESF}}$ feature as described in (4.2). Here, the segments derived in $S$ are also interpreted as interesting segments, which are likely to contain a birds song or call. A modified version of this approach is used for the audio summarisation algorithm proposed in Section 5.2.3.

**Algorithm 1**: Extracts candidate segments for the Savi's Warbler detector. At first, relatively large segments with high energy are extracted. Each iteration, the size of the extracted segments is decreased.

```
Input: $\mathrm{medfilt}_x(n)$ for $n \in \{0, \cdots, N-1\}$ ;          /* median filtered signal */
Output: $S$ ;                                                                  /* segment set */
$s = 5 \cdot 60 \cdot 20$ ;                          /* size of steadiness window:  5 minutes */
$S = \{\}$;
$U = \{(0, N-1)\}$;
while cumulative segment time $< \frac{N}{8}$ do
    calculate $\mathrm{goodpos}_x^s(n)$;
    forall $(f_i, l_i) \in U$ do
        $q = f_i$;
        while $(q < l_i)$ do
            while  $(q < l_i) \wedge (\mathrm{goodpos}_x^s(n) \neq 1)$ do
                $q++$ ;                                            /* skip negative frames */
            end
            $p = q$ ;                                              /* define segment start */
            while $(q < l_i) \wedge (\mathrm{goodpos}_x^s(n) = 1)$ do
                $q++$ ;                                  /* collect new candidate segment */
            end
            if $\left(q - p > \frac{1}{4}s\right)$ then
                $S = S \cup (p, q)$ ;                               /* add segment to output */
                $U = U \setminus^* (p, q)$ ;                /* exclude segment from analysis */
            end
        end
        $s = \frac{3}{4}s$ ;                                       /* narrow steadiness window */
        if $s \leq 12 \cdot 2$ ;                            /* minimal size (12sec.)  reached */
        then
            break;
        end
    end
end
```

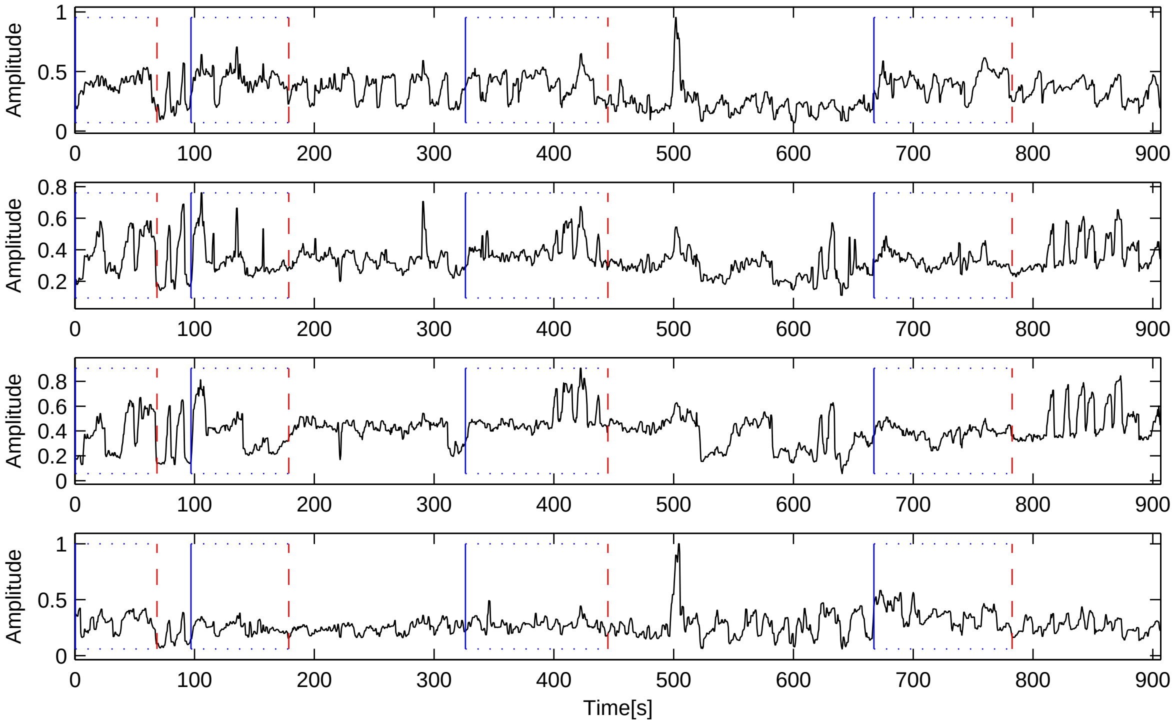


Figure 5.3: Energy curves of a 4-channel, 15 min. duration, monitoring recording. Extracted segments are boxed by blue (start) and dashed red (end) lines.

### 5.2.2 Detection of the canonic Chaffinch's flourish

In the Chaffinch detector, the pre-selection of segments is done by matching the monitoring signal to 15 templates of the Chaffinch's flourish. Signal excerpts being very similar to a flourish template are gathered as flourish candidates for further processing. As the comparison is done in the spectral domain, we use the time anticipating spectral features $F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}}$, as derived in Section 4.1.1, to represent the data during this process. The features are extracted from the frequency band ranging from 2 to 6 kHz. Here, the feature extraction parameters are set to $\#_{\mathrm{bins}} = 40$ coefficients per feature and $\#_{\mathrm{anti}} = 4$ anticipated vectors. The frame rate is set to fps = 50 features per second. In this scenario, the anticipating character of these features is important in order to get hold of the flourish's development over time.

Developing a robust similarity measure on noisy spectral features is a challenging task. Con-

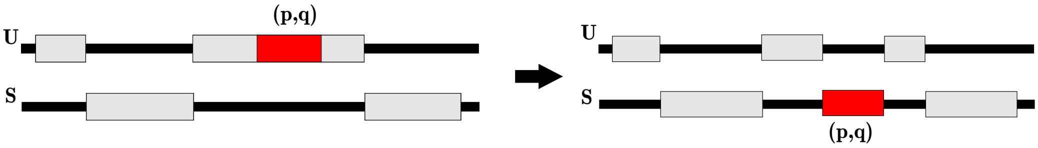


Figure 5.4: Demonstration of the energy segmentation procedure: the sets $U$ and $S$ are depicted before and after the extraction of segment $(p, q)$.

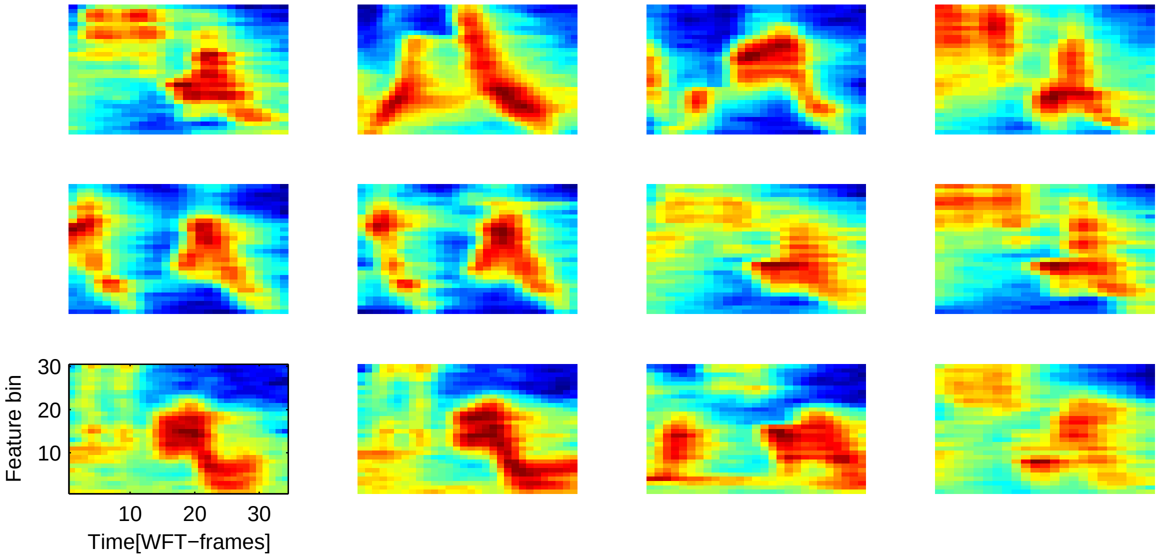


Figure 5.5: $F_{\mathrm{SPEC}}$ features from 12 of the 15 flourish templates, used to gather stanza candidates. Depicted are non-anticipating features instead of the $F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}}$ data used in the algorithm.

sidering the $F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}}$ features, the amount of noise is reduced due to the feature extraction process. As mentioned for the special case of searching self-repeating patterns, using the two-dimensional correlation function is a straightforward way to measure the similarity of chunks of the monitoring signal and the templates. This measure works quite well for searching in a homogeneous data set, where the noise and other signals are either quite steady or silent and the number of singing individuals is limited. As spectral templates can easily be defined by using a graphical user interface such as XBAT, the 2-dimensional autocorrelation measure is a powerful tool for the semi-automatic exploration of such data sets. Given a scenario where full automation and good generalisation over individuals is required, the above measure shows some important drawbacks: at first, given a particular template, the autocorrelation measure is very rigid in defining the temporal sequence of spectrogram features. This leads to disproportionally bad similarity estimates for slightly slower or faster instantiations of the searched call element or phrase. Secondly, special filters and thresholds that are difficult to calibrate a priori are required to suppress background noise in both the template and monitoring signals.

In order to minimize the problems associated with the latter, we use “dirty” samples of the Chaffinch’s flourish, which are extracted from different recordings with diverse background noises. Instead of sticking to the fixed temporary progression given by the sequence of spectral features for each template, this progression is allowed to be edited by means of repetition of feature vectors. Thus locally varying the relation between the template’s and the recording’s progression speed, the resulting similarity measure enables a better-suited comparison for time-scaled instantiations of the flourish template. These are likely to occur, because the length of comparable syllables varies between Chaffinch individuals.

As this “time warping” similarity measure can be efficiently computed using the *Dynamic Programming* strategy, the resulting procedure is commonly called *Dynamic Time Warping*

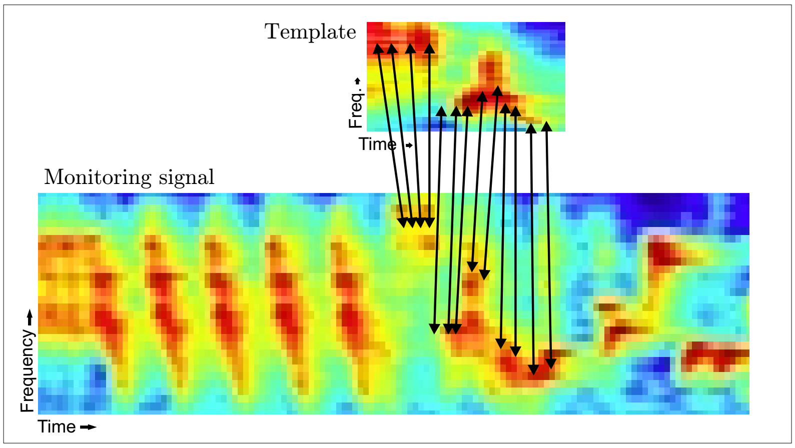


Figure 5.6: Motivation of the time-variable alignment: spectral features of a template and monitoring signal. The arrows connect the corresponding frames of both sequences.

[RJ93]. Formally, the temporal deformation of the feature sequences is represented in form of aligned pairs of $F_{\text{SPEC}}$ feature vectors (see Figure 5.6). Let $X := (x_1, x_2, \cdots, x_M) \in \mathbb{R}^{K \times M}$, for $M \in \mathbb{N}$ be the $F_{\text{SPEC}}$ sequence of a Chaffinch flourish's template, while $Y := (y_1, y_2, \cdots, y_N) \in \mathbb{R}^{K \times N}$, for $N \in \mathbb{N}$ represents an appropriate excerpt of the monitoring signal to be analysed. Here, $K = \#_{\text{bins}} \cdot (\#_{\text{anti}} + 1)$ represents the dimension of the single feature vectors $x_i, y_j \in \mathbb{R}^K$. In order to measure the alignment quality, we first define a local similarity measure $c : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}$ on the $F_{\text{SPEC}}^{\#_{\text{anti}}}$ vectors. In this thesis, for $x, y \in \mathbb{R}^K$, the cosine of the Euclidean angle of two vectors is used:

$$c(x, y) := 1 - \frac{\langle x, y \rangle}{\|x\| \cdot \|y\|}. \tag{5.3}$$

Here, $\|x\|$ represents the Euclidean norm of $x$. In particular, the above measurement is invariant up to the overall energy contained in a feature vector, while measuring the similarity of the frequency distributions of both vectors. Now, the similarity measure corresponding to the entire sequences $X$ and $Y$ depends on the similarity of the sequences induced by the optimal alignment of the two sequences. Vice versa, the term optimal refers to the maximisation of the similarity values derived from the aligned feature vectors. As this cyclic determination may lead to absurd manipulations of both of the sequences, some constraints are incorporated in the process. In order to describe the DTW algorithm and the constraints used to achieve the alignment, we need to formalise the DTW process:

An alignment of two sequences $X, Y$ is given by a *warping path* $p := (p_1, \cdots, p_L)$, where the pairs $p_l = (n_l, m_l) \in \{1, \cdots, N\} \times \{1, \cdots, M\}$ for $l \in [1 : L]$ represent the elementary alignments of the feature vectors $y_{n_l}$ and $x_{m_l}$. For the purpose addressed in this thesis, the following constraints have to be satisfied by a strict warping path:

(i) Boundary condition: $(p_1 = (1, 1)) \wedge (p_L = (N, M))$

(ii) Step size condition: $\forall \ell \in \{1, \ldots, L-1\} : p_{\ell+1} - p_\ell \in \{(1, 0), (0, 1), (1, 1)\}$

Note, that the subtraction of pairs as in (ii) is meant to be component-wise. The above conditions ensure that both of the signals are examined completely in the sense of every template feature vector having at least one associated vector in the tested signal and vice versa. As condition (ii) does not allow steps to past time positions, the warping path's elements are increasing in a weak monotonic fashion. As, a priori, the above optimal alignment is not clear, the similarity measure defined in (5.3) is calcualted for each possible pair of vectors $(i, j) \in \{1, \cdots, N\} \times \{1, \cdots, M\}$, leading to a *similarity matrix* $S \in \mathbb{R}^{N \times M}$, with

$$S_{i,j} := c(x_i, y_j). \tag{5.4}$$

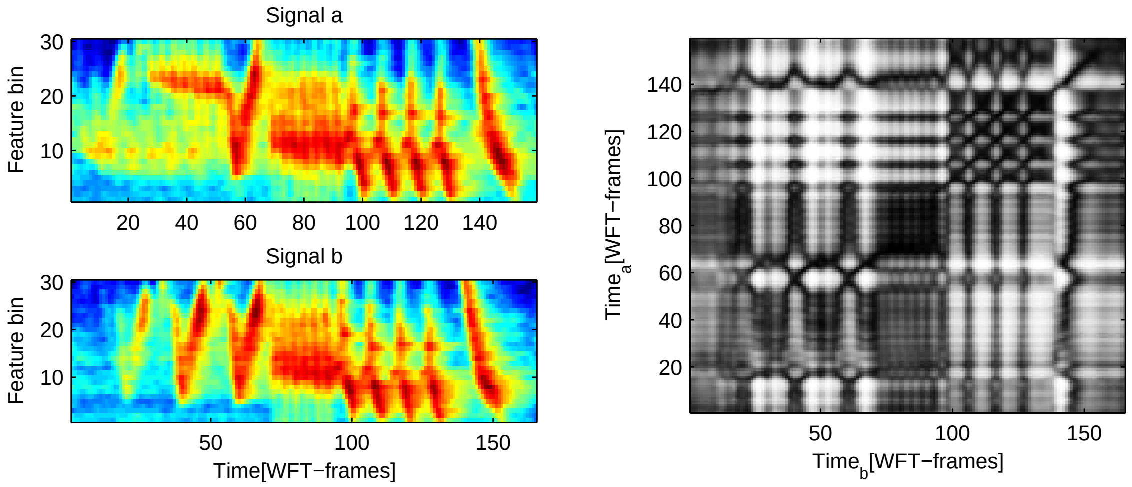


Figure 5.7: Left: spectral features of two chaffinch stanzas. Right: their similarity matrix $S$. Here, black areas depict similar feature vectors.

Calculating this matrix is the most expensive operation involved in the matching procedure, leading to an $O(NM)$ algorithm. There are non-trivial ways to narrow the pairs from which the similarity measure is calculated, leading to faster and comparably accurate measurements. Beneath the use of heuristic restraints keeping the warping path near the diagonal line, a multi resolution approach can be used to define a range of similarity values to be calculated. For further details we refer to [Mat06] and [Ewe07]. In Figure 5.7, such a similarity matrix is displayed for two stanzas of the same individual, differing in their beginnings. As we now visualize this similarity matrix, the motivation of the term "warping path" becomes clearer: imagining the steps $p_l = (n_l, m_l)$ of associated vectors as sequence of positions in the similarity matrix, we can depict a continuous path, starting at the bottom left and proceeding to the top right corner. For general orientation, the lax path, plotted in red in Figure 5.8, may be considered. An optimal warping path is supposed to follow the similarity maxima visible as black ridges in the matrix $S$. The more this path deviates from an approximatively diagonal line, which is clearly visible in the top right corner of the plotted similarity matrix, the more local deviations will be introduced in the final alignment of $X$ and $Y$.

Now, to accomplish the Dynamic Time Warping, we calculate a *warping matrix* $D \in \mathbb{R}^{N \times M}$. During the warping procedure, this matrix will be successively filled with path similarity measures. For each position $(n, m) \in \mathbb{N}^2$, $D_{n,m}$ will contain the similarity measure of the optimal path aligning $(x_1, \cdots, x_n)$ and $(y_1, \cdots, y_m)$. As the step condition (ii) implies a monotonic succession of the time indices to be aligned, the latter matrix can be filled by an algorithm using the following recursion:

$$D(n,m) = \max \begin{cases} D(n-1,m) + c(x_n, y_m), \\ D(n,m-1) + c(x_n, y_m), \\ D(n-1,m-1) + c(x_n, y_m) \end{cases} \tag{5.5}$$

The first row and column of $D$ are initialized to $D(n,1) = \sum_{k=1}^{n} c(x_k, y_1)$, $\forall n \in \{1,\ldots,N\}$ and $D(1,m) = \sum_{k=1}^{m} c(x_1, y_k)$, $\forall m \in \{1,\ldots,M\}$, thus assuming all paths containing the previous positions to begin at position (1,1). Now, the above recursion is applied to fill out $D(n,m)$, for $n \in \{2,\cdots,N\}$, and $m \in \{2,\cdots,M\}$. After completing this computation, it is possible to retrace the optimal warping path: starting at position $(N,M)$, with $D(N,M)$ representing the accumulated similarity measure for the optimal alignment, the path is reconstructed by step-wise following a maximum warping matrix entry, whilst observing the rules given by the step condition (ii) in (5.2.2). For further details on the DTW measure, we refer to [RJ93].

Being designed to align two sequences of approximately equal length, the above warping procedure has to be used with an additional segmentation algorithm, providing appropriate excerpts of the monitoring signal to be compared. A heuristic solution to this, using the $F_{\mathrm{ESF}}$ measure, will be discussed in the following Section 5.2.3. As these procedures "prematurely" define the temporal borders of a sound contained in the monitoring signal, the optimal alignment is rarely achieved in the subsequent aligning procedure. Using a subsequence DTW approach, as described below, will show a way how to use knowledge from the warping matrix to render the task of generating excerpts unnecessary. This is achieved by introducing a more flexible matching procedure. Here, the excerpts will be generated as a by-product of aligning the whole analysis window.

Aligning a template flourish $X := (x_1, x_2, \cdots, x_M) \in \mathbb{R}^{K\times M}$, $M \in \mathbb{N}$, to a large analysis excerpt $Y := (y_1, y_2, \cdots, y_N) \in \mathbb{R}^{K\times N}$, where $N \in \mathbb{N}$ may be 50 times the value of $M$, we have to weaken the path conditions given in (5.2.2) by replacing rule (i) as follows:

(i) Weak boundary condition: $(p_1 = (n_s, 1)) \wedge (p_L = (n_e, M))$, with $n_s \leq n_m$ and $n_s, n_e \in \{1,\ldots,N\}$.

(ii) Step size condition: $\forall \ell \in \{1,\ldots,L-1\} : p_{\ell+1} - p_\ell \in \{(1,0),(0,1),(1,1)\}$

In order to reflect the loosened boundary condition, the initialisation of the warping matrix $D$ is also adapted: while the first row is initialized as explained above, the first column is set to $D(1,m) = c(x_1, y_m)$, $\forall m \in \{1,\ldots,M\}$. The above rules allow the searched path and thus the proposed "matching excerpt" to begin at an arbitrary position within the test signal, ending at an also unrestricted position. Note, that the above condition still requires the flourish template to be fully aligned. In Figure 5.8, the adapted procedure is applied on the Chaffinch stanza's of the previous example. The warping path plotted in the left image starts at position 40, thus leaving the antecedent frames of signal "b" unaligned. Furthermore, considering the warped signals on the right side, we can see how the second element in stanza "b" is used to mimic the weak first element in song "a".

Another notable difference is made through the option of deriving $M$ different valid paths from the warping matrix $D$, corresponding to optimal paths with respect to their ending

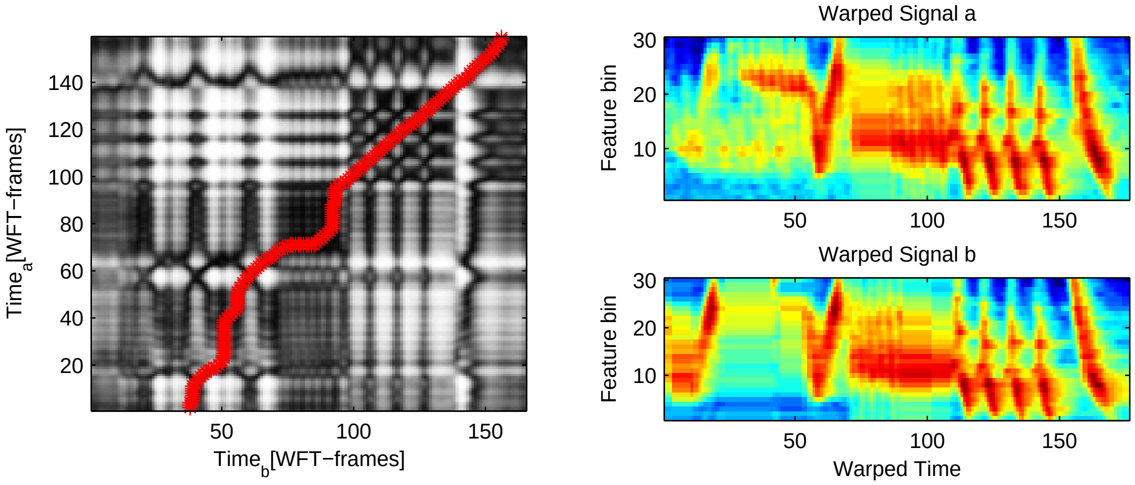


Figure 5.8: Left: similarity matrix $S$ with extracted lax path. The warped spectral feature sequences of the two Chaffinch stanzas are depicted on the right.

point in $Y$. As the values in the last column of $D$ represent the costs of these optimal paths, the vector $D(N,m)$, $m \in \{1,\ldots,M\}$, can be seen as a probability measure for a detected flourish, indicating the plausibility of the actual template flourish ending at position $m$ (and also being performed just before).

By enlarging the excerpt to be aligned as described above, we may expect several Chaffinch stanzas to be included within. Thus multiple paths are extracted from the warping matrix. In Figure 5.9, the extraction of two warping paths is exemplified: Comparing the paths' positions with the associated spectral features, the template shown on the left, is matched to the flourishes of the two visible Chaffinch stanzas. Although the similarity matrix exhibits local maxima for some positions preceding the chosen paths, the last part of the template is only matched at the selected positions. Here, the time-anticipating character of the involved features distinguishes the upwards-downwards progression of the template's second part from the downwards directed progression of the former stanza elements.

As explained in the beginning of this section, noise contained in spectral features such as the $F_{\mathrm{SPEC}}$ features is also affecting similarity measures on these features. Due to the stepwise definition of measure $c$, the derived similarity estimate of a full path is influenced by noisy features. Although an automatic estimation of the number of stanzas contained in the excerpt seems to come with improvements of the overall false positive rate as well as with savings in processing time, thresholding the flourish probability measure is a non-trivial challenge due to the reasons mentioned above. Thus, a fixed number $\#_{\mathrm{matches}}$ of stanza candidates is extracted from each large analysis excerpt, leaving the former task to future work. Using knowledge about the typical inter-stanza intervals performed by Chaffinches, the expected stanza frequency can be estimated for a single individual. As the presence of several birds, singing in call-response structures or even more densely timed, is a common situation, the actual value of $\#_{\mathrm{matches}} = 5$ extracted paths (per 40 seconds and per template) defines a tradeoff regarding *call probabilities*, *false alignments*, e.g. to syllables contained in songs of other birds, the *number of present individuals* and the *number of templates* used.

The alignments are extracted using a greedy two-step algorithm:

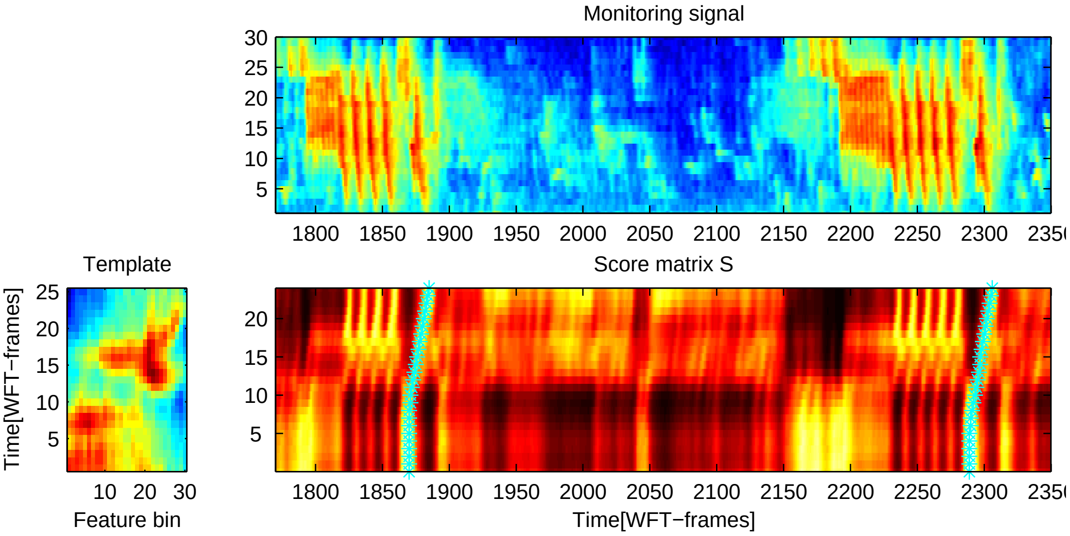


Figure 5.9: Demonstration of the DTW template matching: $F_{\text{SPEC}}$ features of the monitoring signal. The template's spectral features have been plotted with the axes rotated by 90 degrees. Bottom right: similarity matrix, featuring two extracted (cyan) paths, aligning the template to Chaffinch flourishes. Dark/black regions correspond to high similarity values.

1 *Extract* the *path* $p^m = (p_1^m, \cdots, p_L^m)$, where $m = \arg\max_b(D(N,b))$ results the best actual scoring alignment.

2 *Exclude* the aligned time positions from further extraction:
For all $1 \leq l \leq L$, with $p_l^m = (l,k)$, let $D(N,k) = 0$.

Note that for each template the warping matrix $D$ is recalculated and thus the paths excluded in the previous alignment step are again available to the actual alignment procedure. Although picking the best matching excerpts from the monitoring window, the above algorithm is expected to generate many false alignments. Especially when having only few or even no stanzas in the analysed excerpt, many strikingly short false positives will be generated. Thus, in a postprocessing step, segments being shorter than a heuristic, template-based threshold length, are discarded. Actually an aligned excerpt is considered erroneous if it is shorter than half the template's length. In this process, most of the false positives are sorted out. This establishes the length factor of sequences, as valuable quality indicator for the matches derived in the subsequence DTW process. In order to define a ranking of the segments, the associated DTW-path similarity sum is divided by its length and stored,

$$\text{score}(p^m) := \frac{D(N,m)}{L}. \tag{5.6}$$

Finally, having processed all (40 seconds each) analysis windows, the extracted segments are gathered to segment groups or clusters corresponding to simultaneous events by an algorithm to be discussed in the following Section 5.2.3. As the windows are defined in an overlapping manner, an excerpt may be aligned twice, that means at least once per analysis window. Thus, a subsequent segment clustering algorithm, as described in the following chapter, will

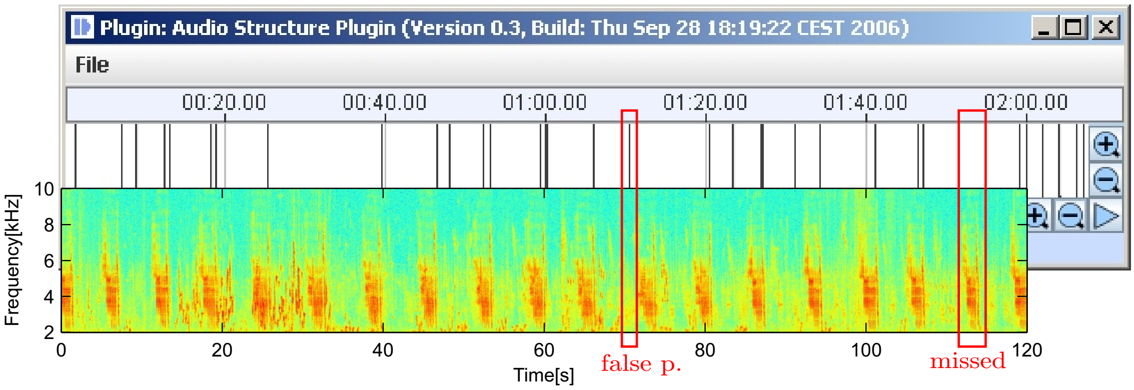


Figure 5.10: Spectrogram of a monitoring signal and SyncPlayer representation of extracted Chaffinch flourish candidates: black lines correspond to individual segments. Note: a grey line is drawn each 20 seconds for better reference.

be able to connect and group the segments extracted in these windows. In the optimal case, the stanza sequences of two Chaffinches, singing in a call- and response manner, are expected to be grouped to two clusters containing the flourish instantiations of the respective bird.

The result of the above template matching and preselection routine can be visualised within the SyncPlayer framework [Fre06]. The graphical user interface of this audio player allows an interactive navigation within particular audio recordings, using the knowledge of several additional data streams such as the annotations gained from the previous template-matching process. In Figure 5.10, the “AudioStructure” plug-in of the SyncPlayer is used to visualise the automatically extracted annotations. In this audio structure window, each of the vertical black lines corresponds to a flourish candidate segment. When comparing the annotated flourish segments with their associated sound in the spectrogram, some false candidates, as the segment associated with the stanza beginning at 70 seconds, can be identified by the reader. Furthermore, some stanzas in the spectrogram have no associated candidate (e.g. at position 110). Being caused by inappropriate additional matches or noisy spectral features, these early classification errors will determine the boundaries for the precision of the final detection.

### 5.2.3 Exploiting stanza repetition (Audio Summarisation)

A useful approach to detect bird voices is to exploit the frequent and, at least for some species, quite exact repetitions of their calls or song stanzas. For this task, the more general *Audio Summarisation* approach seems very promising. The algorithm proposed in the following will form groups of similar audio events occurring in a recording. Thus, given the example of a Chaffinch or Great Tit, singing in the monitoring period we are likely to record several instantiations of their stanzas. Even many species performing a complex and highly variable composition of different stanzas are known to reuse certain blocks at the phrase level. An approach to the identification and grouping of such reoccurring sequences is to be presented in the following paragraphs. From the elements of these groups, a representative one is selected for each group, which is presented to the user or used as an input to a further classification stage. For a set of previously selected frequency bands, the former task is performed separately using four steps:

- *Extract* “interesting segments” using a spectral flatness based criterion.
- For each segment, *analyse* the signal for similar events using DTW. Link events being identified in the same search, including the searched segment.
- *Link* segments according to temporal concurrence.
- *Group* segments having direct or indirect connections to each other.

As the summarisation procedure to be described in the following paragraphs is designed to analyse bird songs or similar acoustical events, the spectral flatness measure, being sensitive to high energy, small bandwidth spectral events, is used to extract the above segments. Based on the $F_{\mathrm{SPEC}}^{\#_{\mathrm{anti}}}$-features used in the Chaffinch’s flourish detector (Section 5.2.2), with parameters set to $\#_{\mathrm{bins}} = 40$, $\#_{\mathrm{anti}} = 4$ and $\mathrm{fps} = 50$, the $F_{\mathrm{ESF}}$ curve is calculated for large analysis excerpts. Unfortunately, in the experiments performed for this work, the length of the analysis excerpts was bound to 40 seconds due to small memory resources on the testing machines. Thus, in the experiments performed with the above parametrisation, events having a repetition interval of more than about 40 seconds were not considered by the algorithm.

The actual segmentation is based on a binary decision vector $b(n)$, indicating the probability of a bird song performed at frame $n$. Let $x(n)$ be the $F_{\mathrm{ESF}}$ feature curve according to the actual analysis window. At first, a moving average filtered version of $x$ is calculated, deriving adaptive mean $F_{\mathrm{ESF}}$ values for a range of about 666 milliseconds: $\overset{s}{\bar{x}}$ for $s = 33$. For each position $n$, the mean and variance values of a neighbourhood $U_n$, maximally containing 20 seconds (1001 coefficients) of the filtered vector, are calculated. Examining an analysis window having a total length of $s_w$ features, for $n \in \{0, \cdots, s_w - 1\}$ we extract a neighbourhood reference vector $R$ as follows.

$$\begin{aligned} R(n) &:= \mathrm{mean}(U_n) + 5 \cdot \mathrm{var}(U_n), \\ \text{with} \quad U_n &:= \left(\overset{s}{\bar{x}}\,(k_0), \cdots, \overset{s}{\bar{x}}\,(k_e)\right), \\ k_0 &= \min(n + 500, s_w) \\ \text{and} \quad k_e &= \max(n - 500, 0). \end{aligned} \tag{5.7}$$

The binary decision vector b is thus defined by

$$b(n) := \begin{cases} 1 & \text{if } \overset{s}{\bar{x}}\,(n) > R_n, \\ 0 & \text{otherwise.} \end{cases} \tag{5.8}$$

Now, a segment extraction algorithm is applied on the decision vector. Besides the requirement of exceeding the minimum length of $\mathrm{minlen} = \frac{\mathrm{fps}}{3}$ samples, or 333 ms, extracted segments have to exceed a threshold of half the binary decisions $b(n)$ being positive in an interval of 600 ms. If a maximum length threshold is not reached, such segments are expanded in a

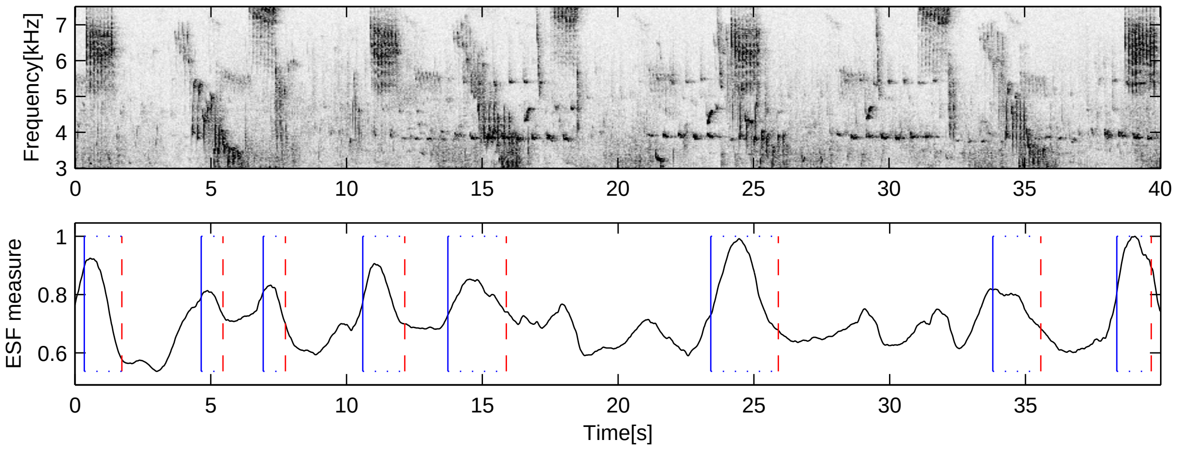


Figure 5.11: Extraction of interesting segments. Top: spectrogram of analysis excerpt $Y$ and Bottom: filtered $F_{\text{ESF}}$ curve. Extracted segments $X^n$ are boxed by blue (start) and dashed red (end) lines.

postprocessing step, gaining up to 20 percent of their length. The actual factor is determined by an examination of the first order derivative of $\overset{s}{\bar{x}}$.

For each interesting segment query $X^n$, such as those marked in Figure 5.11, a DTW search for similar segments is started on the analysis window $Y$. Using the multiple alignment approach as proposed for the Chaffinch's flourish in Section 5.2.2, a fixed number of segments is extracted from the actual analysis excerpt. Finding an appropriate value for the number of aligned segments becomes a complex task if the multiple dependencies on the *number of calls* as well as *calling individuals*, *signal-to noise ratio* and *window length* are taken into account. At this point, deriving $\#_{\text{matches}} = 5$ associated segments per 40 seconds query window will serve as tradeoff answering our purposes. As depicted in the DTW-based search for the Chaffinch's flourish, aligned segments being shorter than half the query's length are discarded at this point.

The above two steps, namely extracting promising queries and searching repetitions of these, are performed on each analysis excerpt. The previous excerpts are designed to achieve a window width of 40 seconds and 50 percent overlap for successive analysis windows. Afterwards, the extracted segments are summarised using a graph algorithm to extract strongly connected components - in short: SCCs - from undirected graphs. The goal of the following procedure is to group segments that

i either are associated to the same DTW-query, or

ii occur simultaneously.

Transporting the segment summarisation task to graphs enables us to make use of some elegant and easy concepts for algorithms on graphs. Basically, graphs are used to model the interconnections between separate entities, called nodes, e.g., $v, u \in V$. Between these nodes, links are defined using the edge set $E$, containing pairs $(v, u), \in E$ of nodes being linked. An undirected graph $G := (V, E)$ is constructed from all segments extracted in the above queries. For a total number of $N$ queries $X^0, \cdots, X^{N-1}$, each extracted warping path $\widetilde{P^{k,n}}$, $k \in \{0, \cdots, \#_{\text{matches}} - 1\}$, $n \in \{0, \cdots, N-1\}$ is associated to a node $v_{k,n}$. As the summarisation is performed on the whole recording, the original paths $P^{k,n}$, containing window-based references, are transformed to absolute frame positions, yielding $\widetilde{P^{k,n}}$. Moreover, $n$ specifies the query from which the respective segment results. Now, the edge relation $\sim$ is defined according to the conditions (i) and (ii):

$$v_{j,m} \sim v_{k,n} \Leftrightarrow [m = n] \vee \left[\text{shrdtime}\left(\widetilde{P^{j,m}}, \widetilde{P^{k,n}}\right) > \frac{1}{2}\right]. \tag{5.9}$$

Figure 5.12 depicts the envisaged graph representation of an analysed monitoring excerpt. Here, a single edge is justified by condition (II), leaving the remaining edges to (I). For $P^{k,n} = (p_1, \cdots, p_K)$, $P^{j,m} = (q_1, \cdots, q_L)$, $C := \max(K, L)$ and $C, K, L \in \mathbb{N}$, the function shrdtime measures the time shared by the two paths associated with the above nodes:

$$\text{shrdtime}(P, Q) := \frac{1}{C}\, |\{j \mid \exists g, h, v, w \in \mathbb{N} : p_g = (v, j),\, q_h = (w, j)\}|\,. \tag{5.10}$$

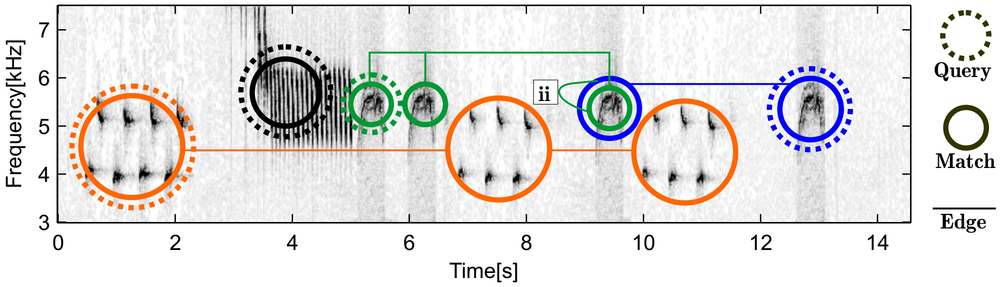


Figure 5.12: Demonstration of an audio summarisation graph. The example signal, being constrained to 3-7.5 kHz, is constructed from calls of the Great Tit (*Parus mayor*, orange), Common Swift (blue and green) and the Blue Tit (*Cyanistes caeruleus*, black).

The edge set $E := \{(v_{j,m}, v_{k,n}) \mid v_{j,m} \sim v_{k,n}\}$ results from the direct application of the above relation. Now, *strongly connected components* are extracted. *SCCs* are defined as maximal subgraphs $S_k := (V_k, E_k)$ of the graph G, where the nodes $v_j^k \in V_k$ are completely linked. For the above example, there are two SCC's, containing the big orange nodes (Great Tit) on the bottom in the first SCC while combining the upper blue and green nodes (Common Swift) in a second component. Formally, for each pair $v_j^k, v_0^k \in V_k$ there is a sequence of $l + 1$ edges $\left((v_j^k, p_0), \cdots, (p_l - 1, p_l), (p_l, v_0^k)\right) \in E_k^{l+1}$, connecting the above nodes.

There are lots of implementations for the task of extracting SCCs from graphs, including linear time ($\Theta(|V| + |E|)$) algorithms. For the proof of concept targeted in this work, a naive *union-find* implementation is used. As the usual number of nodes and edges is quite small ($< 1000$), the computational costs of the naive implementation become irrelevant in comparison to the expensiveness of the feature extraction step. For a detailed discussion of the relevant algorithms on graphs, we refer to [CLRS01].

The extracted SCCs are interpreted as signal groups: the segments associated with the contained nodes are expected to have similar content. Unfortunately, the concurrence measure used for the definition of the edge relation in (5.9) does not check the acoustical similarity of the segments to be connected. Hence, erroneous links are e.g. produced by segments featuring the simultaneous calls of two birds. As the number of matches returned for each query is fixed, some of the erroneous alignments being put out to fulfil this condition may also cause false groupings. Thus, the $\#_{\text{matches}}$-parameter is a useful value to adjust when tuning the summarisation procedure. Generally, the proposed algorithm is likely to work well on recordings with a bounded amount of crosstalk in the analysed frequency bands. When working with a large set of signal sources, the setup of the individual analysis bands is crucial.

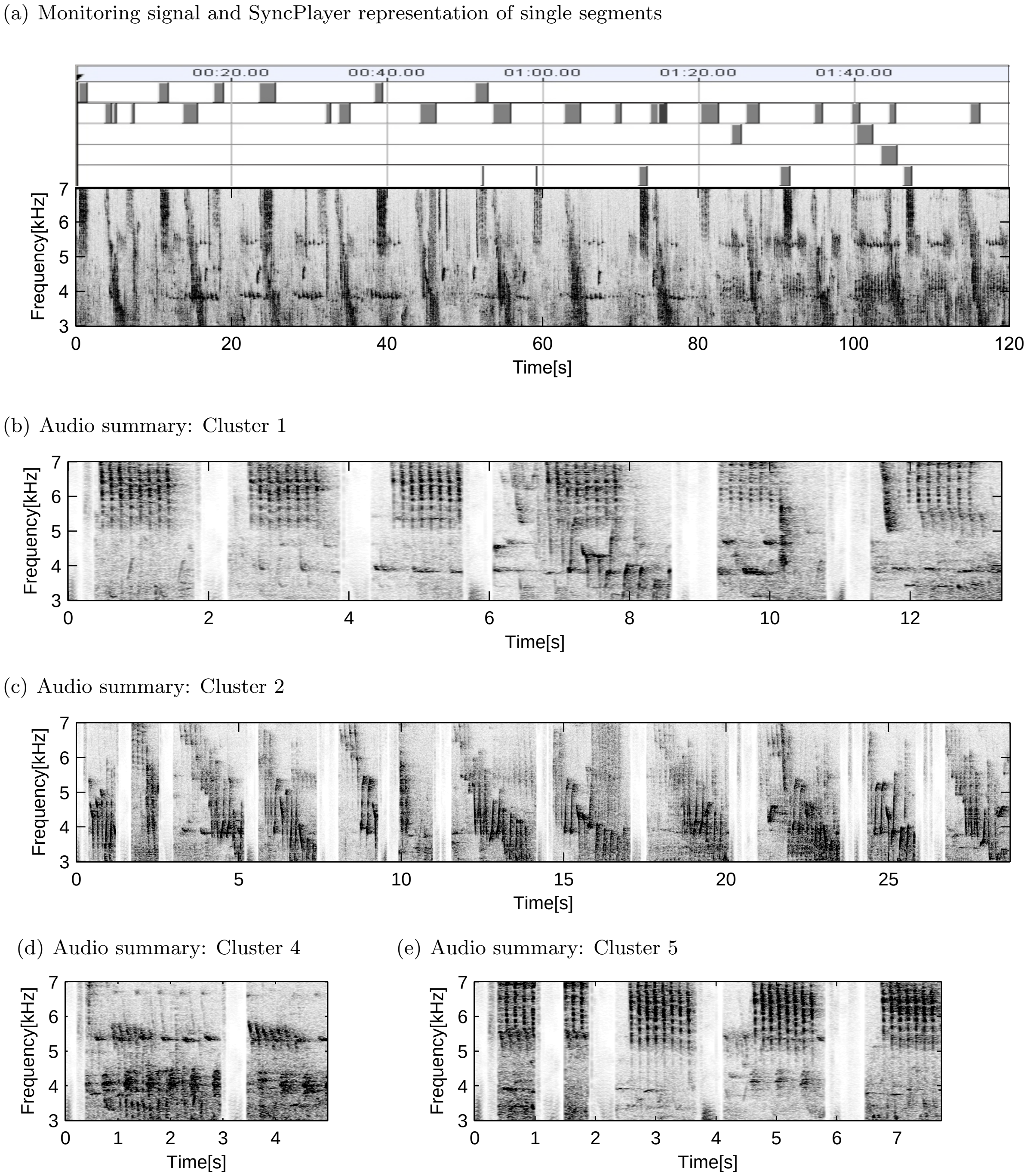
As for the template matching procedure, annotations in a SyncPlayer compatible output format are generated, containing the extracted segments and the cluster structure. Furthermore, an audio representation is derived for each cluster by concatenating the associated audio excerpts. Thereby, the individual segments are sorted according to their DTW-similarity score. A human voice, announcing the segments' ranks, interleaves each pair of segments. In Figure 5.13, the introduced representations of the summarisation result are depicted: a monitoring signal was analysed, focussing on the 3-7 kHz frequency band. The number of rows being displayed in the SyncPlayer table corresponds to the actual number of 5 remaining clusters. Comparing the spectrograms of cluster 1 and 5, the two look very similar. Indeed, because of the exclusive two common segments (at about 52 seconds) largely differing in their compared size, the two parts of the specific bird's song were not joined in the summarisation step.

However, a representative segment may now be chosen for each derived cluster. Besides picking at random, as performed in this work, both the temporal position and length of a segment may be used to define a segment ranking within each cluster. An even more sophisticated way would be to compute the DTW-similarity for all pairs of segments in the particular cluster. For each segment, a score may be defined through its summed similarity to the rest of the group's segments.

For bird detection purposes like the automatic censusing of species occupying a certain habitat, the classification of the above representatives may lead to a useful and cheap summary of the whole recording. Of course, the classification results gathered in this first step can be improved using the remaining segments in the respective group.

## 5.3 Song detection using periodicity features

In the following sections, the periodicity features and several derived measures, introduced in Section 4.2, will be used to detect the Savi's Warbler and other species having a song of a similarly constant periodicity. In this case, the $F_{\text{NPS}}$ features show a very high robustness to ambient noise while providing a basis for an effective analysis of such songs. The next section is dedicated to the specific application of the mentioned features on the detection of the Savi's Warbler. In order to depict the general potential of the $F_{\text{NPS}}$ features and their relatives, Section 5.3.2 discusses a more generic representation of these, opening the field of the features' applications towards multiple class classifiers, which could be used for the detection of several other species.


(a) Monitoring signal and SyncPlayer representation of single segments

(b) Audio summary: Cluster 1

(c) Audio summary: Cluster 2

(d) Audio summary: Cluster 4

(e) Audio summary: Cluster 5

Figure 5.13: Audio summarisation demo: SyncPlayer table showing the segments of the 5 automatically extracted clusters, spectrogram (a) of the analysed band of a monitoring recording and audio summaries (b-e) of the individual clusters. Note: segments are sorted by rank in audio summaries. As the 3th cluster is singular, the waveform is not depicted. Cluster number 2 (plot c) was truncated to retain the remaining segments' fine spectrogram structure. Separating the single segments, the human voice announces each segment's number.

### 5.3.1 Savi's Warbler: Characteristic period recognition

In this chapter, a procedure to detect very static, periodic birdsongs will be presented, using the Savi's Warbler's song as an example. In the design process of the following approach, special attention was put on the robustness required for working with signals recorded in noise conditions typical for unsupervised monitoring. As explained in Section 4.2.6, the $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features, defining the basis for this classifier, are well suited for the prospected scenario. In Figure 5.14, the Warbler's signal on the left is almost imperceptible due to the simultaneous noise. Still, the depicted $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features indicate the Warblers repetition frequency.

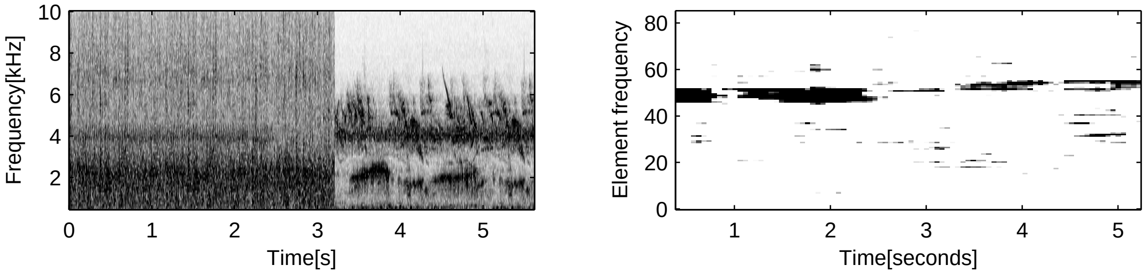


Figure 5.14: Left: noisy recordings of two Savi's Warbler songs. The first part of the monitoring signal contains heavy rain noise. The second half of the signal is dominated by the overlapping songs of many birds. Right: associated $F_{\mathrm{NPS}}$ features.

Now, either for the candidate segments $S$, derived in Section 5.2.1, or for the whole monitoring signal, several one-dimensional measures will be applied, indicating the following properties of a periodic signal:

i Autocorrelation sharpness ($F_{\mathrm{ACSHARP}}$)

ii Dominant element repetition frequency

iii Relative strength of this frequency (compared to other, simultaneous frequencies).

At first, autocorrelation features are extracted for 5 subbands of the Warbler's typical band (3.5-4.8 kHz) and a flanking band (0.5-1.5 kHz), using spectral features with fps = 300 frames per second and $\#_{\mathrm{bins}} = 30$ bins. The resulting autocorrelation features, having a maximal lag of $s_{ac} - 1 = 127$ WFT-frames, are sensitive to periodics in a range of approximately 2.4 to 150 Hz. These features are calculated each $\Delta_{ac} = 16$ WFT-frames, leading to a total of $\mathrm{fps}_{\mathrm{ac}} = \frac{\mathrm{fps}}{\Delta_{ac}} = 18.75$ ACORR-frames per second.

The main classification criterion will ensure the frequency (ii) to remain in the range expected for a Savi's Warbler's song, which is bounded by 44 and 60 Hz. Although, in a previous version of the algorithm, the original $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features were used for the extraction of the above values, we will introduce a slightly modified version of these features. In Section 4.2, the $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features are derived from the subband $F_{\mathrm{ABP}}$ features. Here, a best subband, containing the clearest element repetition, is selected, which then represents the whole analysed band. As, in this particular case, the algorithm is designed to search for a distinct element repetition frequency, the above features fail to serve this purpose given a simultaneous and stronger element repetition with a different repetition frequency. In this case, the autocorrelation -

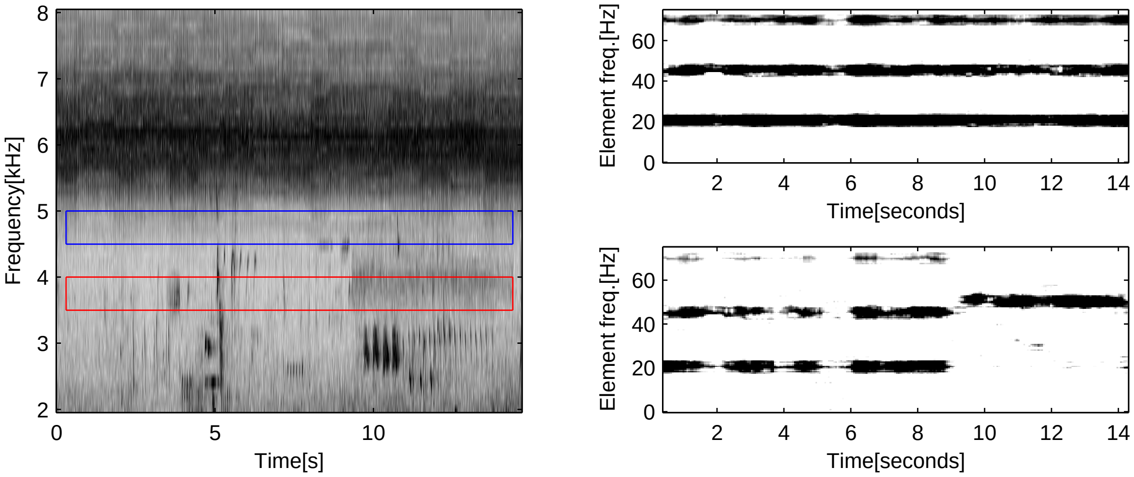


Figure 5.15: Left: spectrogram featuring a loud Grasshopper Warbler's (*Locustella naevia*) song, and a weak Savi's Warbler's song at 10 seconds. Boxes depict the range of two subbands. Right: $F_{\text{NPS}}^{-\text{nois}}$ features of the respective frequency bands in the same order.

and thus the $F_{\text{NPS}}^{-\text{nois}}$ features - will be focused on that second signals frequency, more or less eliminating the relevant signal.

In Figure 5.15, the dominant Grasshopper Warbler's song is masking the searched bird's song. Still, in the lowest subband, the Savi's Warbler's typical 50 Hz dominate the Grasshopper's 22 Hz and their harmonics. Thus, instead of selecting a best subband for each ACORR-frame, as described in Section 4.2.4, Equation (4.18), all of the 5 subband autocorrelation sequences $F_{\text{ACORR}}^{\text{norm}}[m]$, $m \in \{0, \cdots, 4\}$ are examined separately. Unfortunately, the amount of data to be evaluated has grown by a factor of 5. In order to compensate for this, in each feature extraction stage of the following procedure, subband-frames not fulfilling some particular conditions on the calculated features will be excluded from further analysis. To keep track of the subband frames being queued for further processing, the set $R_s = \{r_0, \cdots, r_K\}$, with $r_k = (m, n) \in \{0, \cdots, 4\} \times \mathbb{N}$ will specify these frames by band number $m$ and time position $n$. $R_0$ is initialized containing all subband frames $R_0 := \{(0,0), \cdots, (\#_{\text{sbands}} - 1, L - 1)\}$, where $L \in \mathbb{N}$ represents the length of the $F_{\text{ACORR}}^{\text{norm}}$ sequence derived from the segment to be classified. In this stage, the autocorrelation sharpness is calculated for each subband frame $r_k$. Frames with corresponding sharpness values missing a fixed threshold of 0.1 are filtered out.

$$R_1 := \{(m, n) \in R_0 \mid F_{\text{ACSHARP}}[m](n) > 0.1\}. \tag{5.11}$$

Here, $F_{\text{ACSHARP}}[m](n)$ refers to the autocorrelation measure defined in (4.16), being applied on the $m$-th subband. Now, for the remaining frames, the denoised novelty power spectrum features $F_{\text{NPS}}^{-\text{nois}}[m]$ are extracted, using a DFT of length $N = 512$ and a flanking band, as defined above, for the denoising process. As explained in Section 4.2.6, the autocorrelation curves are extended to the DFT-length using zero-padding.

For each such subband-frame $r_k \in R_1$, the dominant element repetition frequency is estimated by picking the maximum value of a truncated version of the novelty power spectrum features. As the $F_{\text{NPS}}^{-\text{nois}}$ features usually carry much energy in the coefficients corresponding to low

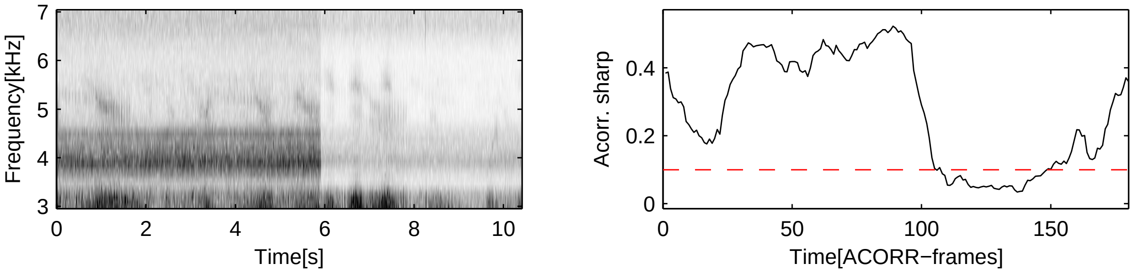


Figure 5.16: Left: spectrogram featuring a Savi's Warbler's song. Right: $F_{\text{ACSHARP}}$ features of a single subband with dashed threshold line.

frequencies, the maximum search starts from an index $\omega_0$, associated to the frequency of 15 Hz, thereby omitting the latter critical coefficients. A frequency estimate is then defined by

$$\omega[m](n) := \frac{\text{fps}}{N} \arg\max_{w>w_0}(F_{\text{NPS}}^{-\text{nois}}[m](w,n)), \tag{5.12}$$

for $(m,n) \in R_1$, given a $\text{DFT}_N$ for the calculation of the novelty power spectra $F_{\text{NPS}}^{-\text{nois}}[m]$. Again, the set of subband-frames is filtered, retaining the segments featuring pitch estimations in a range of 44-58 Hz.

$$R_2 := \{(m,n) \in R_1 \mid 44 \le \omega[m](n) < 58\}. \tag{5.13}$$

Now, the remaining subband-frames are measured for the "dominance" of the Warbler's periodicity range. Therefore, the energy contained in the coefficient of the novelty spectrum, corresponding to $\omega$ is compared to the overall energy in the spectrum. Actually, the measured energy, not directly corresponding to the energy of the acoustic event, indicates the relative strength of the periodic series of novelty peaks. Thus, the actual comparison is done regarding a feature similar to the autocorrelation sharpness measure. Let $p = 75, q = 99$ define the novelty spectrum positions corresponding to the frequencies of 44 and 58 Hz. The domination feature $F_{\text{dotn}}[m](n)$ is then defined by

$$F_{\text{dotn}}[m](n) := 1 - \frac{E_{(m,n)} - \sum_{f=p}^{q} F_{\text{NPS}}[m](f,n)}{E_{(m,n)}}, \tag{5.14}$$

$$\text{where} \quad E_{(m,n)} = \sum_{f=0}^{N/2-1} F_{\text{NPS}}[m](f,n).$$

High $F_{\text{dotn}}$ measurements indicate a low percentage of simultaneous periodic signals. Before classifying the signal, the time resolution of the above features is reduced to a sample rate of 4 Hz by means of grouping blocks of successive features into time bins $B^k$. Furthermore, the individual subband results are summed into a shared bin. Thus, the $F_{\text{dotn}}$-feature values

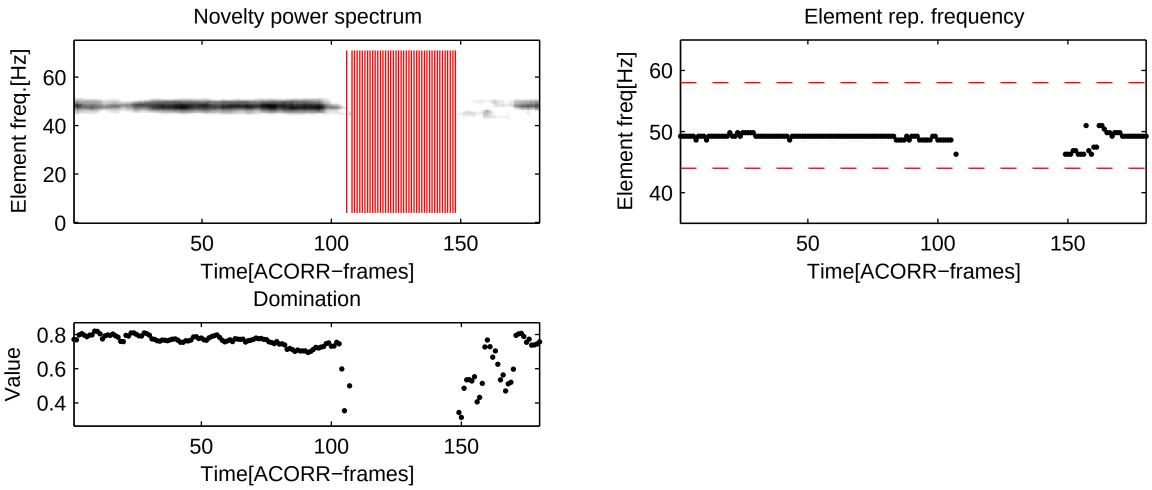


Figure 5.17: Continued example from Figure 5.16, depicting features from a selected subband: Top left: novelty power spectrum features derived for frames in $R_1$. Omitted frames are marked by red lines. Top right: element repetition frequency estimates $\omega[m](n)$ and frequency thresholds as dashed lines. Bottom: $F_{\text{dotn}}[m](n)$ domination feature for frames in $R_2$.

corresponding to the same bin are summed up, and saved as $F^*_{\text{dotn}}$.

$$F^*_{\text{dotn}}(k) := \begin{cases} 0, & \text{if } B^k = \{\}, \\ \sum_{(m,n)\in B^k} F_{\text{dotn}}[m](n), & \text{otherwise,} \end{cases} \tag{5.15}$$
$$\text{where} \quad B^k := \left\{ (m,n) \in R_2 \,\middle|\, k = \left\lceil 4n \frac{\Delta_{ac}}{\text{fps}} \right\rceil \right\}$$

Time bins lacking corresponding frames are set to zero. Another feature is generated by counting elements in $R_2$ which contribute to a particular time bin. Subsequently, the curve is smoothed by applying a sliding mean filter having a length of two seconds or eight samples.

$$F_{\text{warb}}(k) = \overset{8}{\overline{\left(F\tilde{}_{\text{warb}}(\cdot)\right)}}(k), \quad \text{where} \tag{5.16}$$
$$F\tilde{}_{\text{warb}}(k) := |B^k|.$$

When analysing a multi-channel recording, this complete above procedure is performed for each channel separately. At this point, the results are combined by frame-wise summation of the respective $F_{\text{warb}}$ curves. The $F^*_{\text{dotn}}$ curves are summed in the same manner. Now, a preliminary segmentation is performed: maximal segments $o_j \in O$ are extracted, fulfilling the following treshold requirements on the previous measurements:

$$O := \{(p,q) \in \tilde{O} \mid \nexists (v,w) \in \tilde{O},\, v \le p, q \le w :\, w - v > q - p)\}, \tag{5.17}$$
$$\text{where} \quad \tilde{O} := \{(p,q) \mid \forall p \le l \le q : (F_{\text{warb}}(l) \ge T_w) \wedge (F^*_{\text{dotn}}(l) \ge T_d)\}.$$

The above thresholds are usually set to $T_w = 2$ and $T_d = 5 \cdot \frac{\#\text{channels}}{2}$. Afterwards, given a minimal segment interspace of 3 seconds, closely grouped segments in $O$ having an interspace smaller than 3 seconds are combined and replaced by a summarising segment. Finally, segments having a total length of less than 3 seconds are discarded.

This segmentation routine finalises the classification procedure, labelling all remaining segments as songs of the Savi's Warbler. These segments are now ranked according to the mean $F^*_{\text{dotn}}$ value, which is computed for each segment. Using a suitable software for playback of annotated audio, like the SyncPlayer [SYN] or Audacity [AUD] software, the results of the above classification can be browsed in an intuitive manner.

When operating with multichannel recordings, the above procedure can be extended to estimate a location of the sound source. This is achieved by comparing the $F^*_{\text{dotn}}(k)$ values of the different channels. The channel provided by the microphone being located nearest to the source is likely to gain the highest feature value. This is particularly caused by the minimal signal to reverb and signal-to noise ratios provided by the selected microphone. In Figure 5.18, this kind of estimation is illustrated using the Audacity software. Being capable of displaying the 4 channels of the monitoring recordings as well as generic textual annotations, this software is used for the representation of the detector's output. Fortunately, the required annotation format is plain ASCII text, thus, the produced detection results can be easily integrated into further applications.

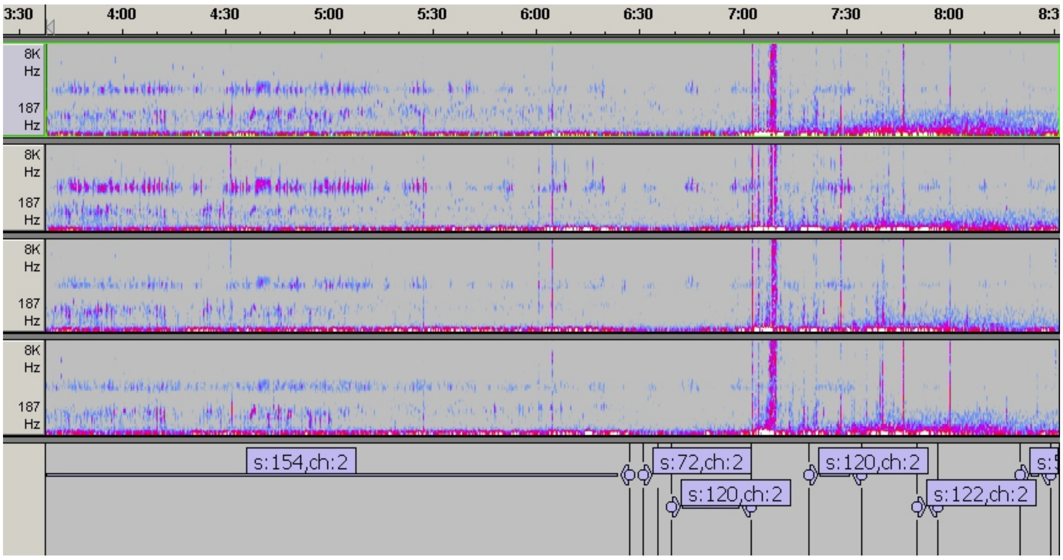


Figure 5.18: Audacity software displaying a 4-channel spectrogram of a monitoring record. Additionally, the result of the Savi's Warbler detector is plotted in the bottom graph. Textual annotations follow the format "s:score, c:estimated channel".

### 5.3.2 Classification based on element frequency

Considering the robustness and accuracy of the above algorithm, which is evaluated in Section 6.1, the adaptation of the Savi's Warbler detection algorithm and its features to detect further species is seems very promising. Typical target species are other members of the *Locustella* family, namely the Grasshopper and River Warbler, as well as many crickets and some toads. Instead of tuning the features introduced in Section 5.3.1 to the particular properties of each species, a general approach is motivated in this chapter. By means of extracting several additional parameters from the previous features, the enlarged feature set is designed to serve as the base data for classifiers supporting unsupervised training. Thus, leaving the task of defining accurate classification boundaries to an automatic process, the task of expanding the features' application range is subject to the availability of training data sets for distinct species, reducing the manpower needed to perform the tuning.

Similar to the feature extraction applied for the Savi's Warbler's detection, a set of subband novelty power spectrum ($F_{\text{NPS}}$) features is derived for a set of subbands. In this generic approach, the frequency range of 1.5-10 kHz is split into 29 half-overlapping frequency subbands. As the actual frequency band of an envisaged species in not known in advance, no flanking band is defined and no denoising is applied on the features. Based on spectral features being computed at fps = 300 frames per second, with $\#_{\text{bins}} = 150$ bins, the 29 subbands' ACORR-frames of length $s_{ac} = 128$ are extracted each 64 WFT-frames.

Basically, the dominant repetition frequency is extracted from each of the 29 $F_{\text{NPS}}$ sequences. This is achieved by the application of the element repetition frequency extraction routine used in the Savi's Warbler detector (see (5.12)). The frequency estimations extracted in this step are improved on a frame by frame basis. This is achieved by an algorithm using the local $F_{\text{NPS}}$ maxima to verify or correct the pitch estimations. Integrating the frequency ranges associated with the respective subbands, a set of triples $(\omega[m](n), f_m, n) \in \{0, \cdots, 127\} \times \mathbb{R} \times \mathbb{Z}$, containing the detected repetition frequencies with the associated subband's center frequency $f_m$ and ACORR-frame $n$, is extracted. Here, an optional filtering step, excluding frames associated to minor autocorrelation measurements, may be applied. The resulting features were, in a version modified to guarantee acceptable computation times, integrated into the interactive web interface [TSA] of the Animal Sound Archive at the Humboldt-University Berlin, enabling the users of this distributed sound database to immediately access the element repetition frequencies of various species, by means of a "periodicity measurement filter" (see Figure 5.19).

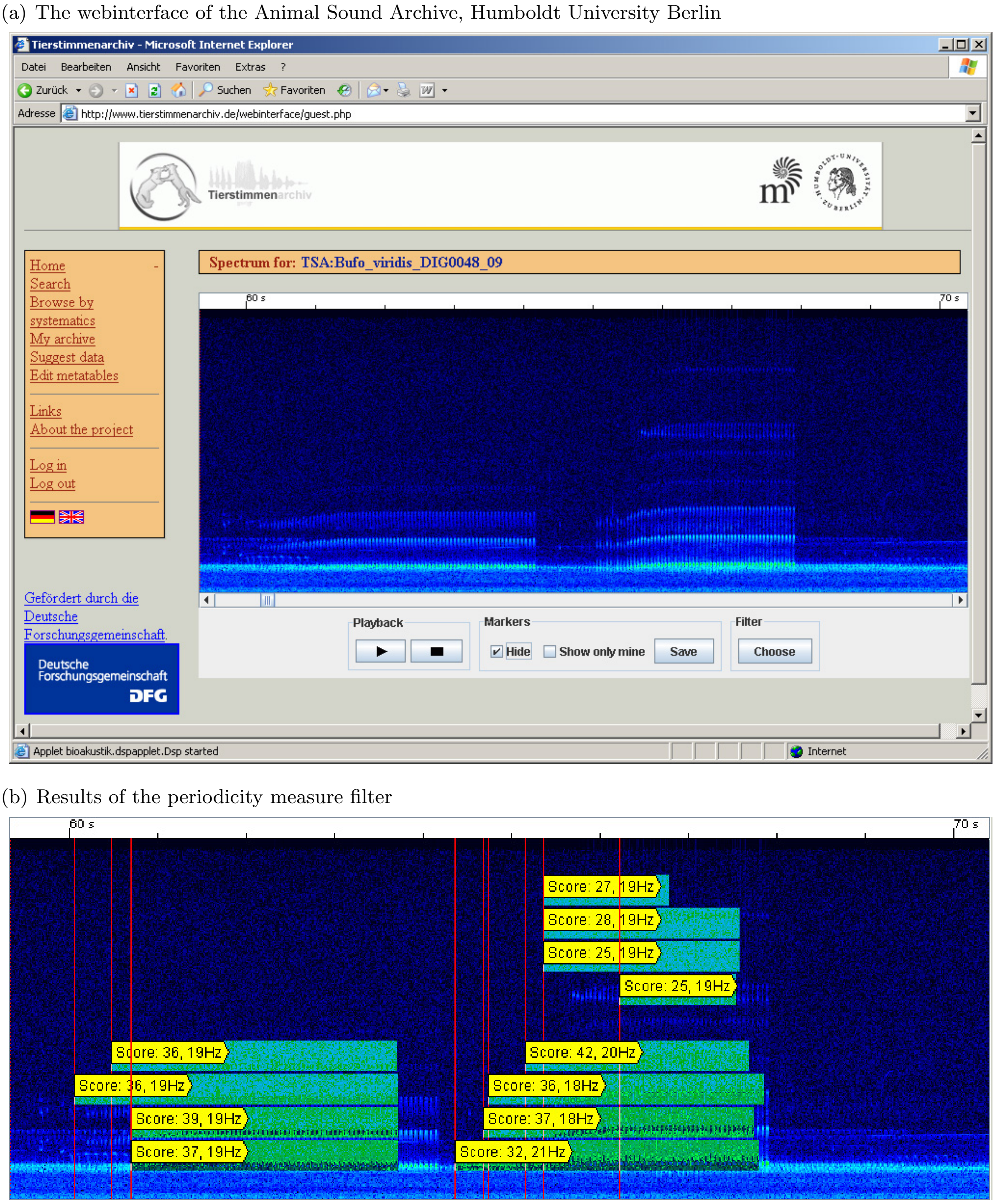
In the given example, in addition to the repetition frequency, a score is indicated for each periodic segment. This score is computed using the domination feature $F_{\text{dotn}}$, as depicted in figure 5.17. The $F_{\text{dotn}}$ measure may be integrated by enhancing the above triples to quadruples. Now, configurations of subband-periodics, being extracted at the same ACORR-frame, can be analysed: let

$$M_n := \{m_i^n \mid F_{\text{ACSHARP}}[m](n) > T_1\}$$

contain the indices of all subbands fulfilling an autocorrelation sharpness condition with threshold $T_1$, then a series of feature configurations

$$F_{\text{abstractp}_n} := \begin{pmatrix} \omega[m_1^n](n) & f_{m_1^n} & F_{\text{dotn}}\left[m_1^n\right](n) \\ \vdots & \vdots & \vdots \\ \omega\left[m_{|M_n|}^n\right](n) & f_{m_{|M_n|}^n} & F_{\text{dotn}}\left[m_{|M_n|}^n\right](n) \end{pmatrix}, \text{ for } n \in \mathbb{Z} \tag{5.18}$$

can be defined. Note that the above feature represents both the frequency bands used by the calling or singing animal as well as the individual element repetition frequencies produced



(a) The webinterface of the Animal Sound Archive, Humboldt University Berlin

Tierstimmenarchiv - Microsoft Internet Explorer

Adresse http://www.tierstimmenarchiv.de/webinterface/guest.php

Spectrum for: TSA:Bufo_viridis_DIG0048_09

Applet bioakustik.dspapplet.Dsp started

(b) Results of the periodicity measure filter

Figure 5.19: (a) The web interface of the Animal Sound Archive, showing a spectrogram of the European Green Toad (*Bufo viridis*), delivered through a java applet. In the bottom image (b), markers, set by means of an approximative online-implementation of the $F_{\text{abstractp}}$ features, label periodic sequences by their inherent element repetition rate (in Hz). Furthermore, the mean $F_{\text{dotn}}$ value is noted as score for each such segment.

thereby. Furthermore, the $F_{\text{dotn}}[m](n)$ measurements may be used as a weighting function defining the importance of the associated repetition frequency $\omega[m](n)$. Now, given a set of recordings for a distinct species, a modelling approach supporting unsupervised learning techniques can be followed. Once the $F_{\text{abstractp}_n}$ feature series is extracted, the contained vectors can be reshaped to achieve a constant dimension, by narrowing the contained subbands to those which are shared by the majority of all data points. Now, for example, a Neural Network may be trained on the basis of the narrowed features. The proposed scheme may be extended to a set of species, obtaining a multiple-class classifier.

## 5.4 Chaffinch: Model based song detection using extracted structures

The following sections are focused on the extraction of structures inherent to the song stanzas of some particular bird species. An energy-based procedure for the separation of stanzas into phrases and/or elements was developed and tested by Fagerlund [Fag04], using a testing set of relatively clean recordings. As the goal of this thesis is to robustly extract such structures from monitoring material having low signal-to noise ratio's, energy based measures were considered to lack the necessary amount of robustness. Reducing the set of desired target species to those emitting song stanzas containing repetitive phrases (see Chapter 2.4.1), the periodicity based features can be used to approach this task. A first application of these features, used as second stage for the detection of the Chaffinch, constitutes an element repetition frequency based procedure for structure extraction. Section 5.4.2 will present the use of some standard vector-clustering methods for the segmentation of autocorrelation sequences, as introduced in Section 4.2.2.

### 5.4.1 Model evaluation based on dominant periods

Continuing the description of the Chaffinch detector from Section 5.2.2, a basic stanza segmentation approach will be discussed in this section. Before classifying the candidate segments derived in the aforementioned chapter, these segments are sorted in a descending order, according to their DTW-similarity ranking. Now, beginning with the topmost segment, each segment is processed separately. Operating on a signal excerpt being located before the respective flourish candidate, a segmentation into phrases is attempted. During the segmentation procedure, two periodicity features, derived from autocorrelation curves, are of importance: a pitch estimate is used to define areas of elements repeating within a nearly constant period. Thereby, the autocorrelation sharpness measure, as defined in (4.16), is used to validate the significance of the estimated periods. The extracted segments, featuring strong and constant periodic repetitions, are treated as phrases in a birdsong, as defined when introducing the Chaffinch's song structure (Sect. 2.4.1). Then, by means of applying a set of rules on the segment level, particularly for

- segment interspace,
- segment length,
- the inter-segment variance of periods and

- a total number of segments,

a decision on whether a Chaffinch's stanza was being analysed is provided.

**Preliminary model check.** Solely representing the tail of a Chaffinch's song, the segments, given as input to the classification stage, do not contain sufficient data to classify a full stanza. Assuming a maximum stanza length of 3.8 seconds, neglecting the flourish's length, features have to be extracted for an excerpt of this length. Incorporating a first bit of prior model knowledge, the excerpt or "stanza-body", is chosen to end at the beginning of the candidate segment, thus defining the stanza to end with the respective flourish. To reflect the approach intended by the following procedure, the enumeration of segments and time spans shall be understood in a reverse manner, thus starting at the last element of the stanza-body, before the flourish. Then, the process proceeds - in reverse temporal order - towards the first elements.

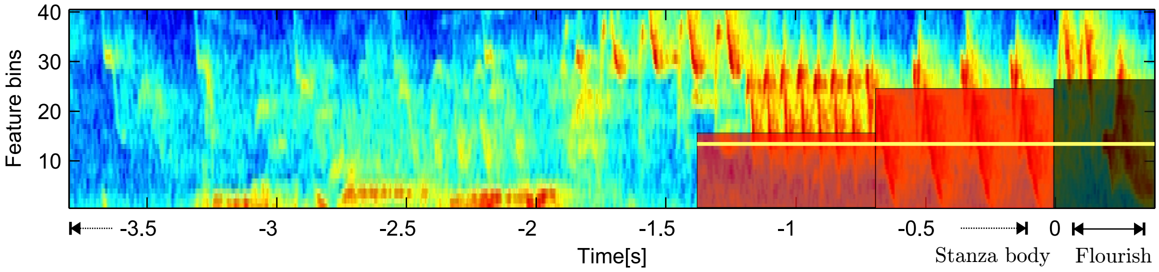


Figure 5.20: Spectral features for a stanza-body candidate (negative time values) and associated flourish segment (positive values). Boxes illustrate average energies. A yellow line marks the "quick check" energy threshold.

As the following segmentation procedure is computationally expensive, a "quick check" is made in advance, verifying a second and necessary classification criterion: the duration of the stanza-body is required to be at last 1.3 seconds. This requirement is assured by extracting a mean energy value of the Chaffinch's typical band (2 - 6 kHz) for two signal blocks of 0.6 seconds, splitting an 1.3 seconds excerpt afferent to the stanza-tail (see red boxes in Figure 5.20). Each of these mean values has to exceed half the mean energy value calculated for the suspected stanza's flourish. In Figure 5.20, this threshold is indicated by a line. The first of the previous blocks usually covers a small gap between the expected stanza-body and flourish. A low energy value measured for this block indicates a gap of excessive length. Thus, some attention is given to ill-conditioned cases featuring solitary flourish candidates drawn from, or accompanied by calls or songs of other birds. Failing the energy test, a segment is discarded from further analysis.

**Period extraction** After passing the above test, in order to perform the element repetition period and autocorrelation sharpness measurements, normalised autocorrelation ($F_{\mathrm{ACORR}}^{\mathrm{norm}}$) features are extracted for the actual excerpt. Here, the initial analysis is focussed on the frequency band between 2 and 6.5 kHz. With the parametrisation reading a maximum lag of $s_{ac}$ = 81 WFT-frames and a very small stepwidth of $\Delta_{ac} = 1$, the ACORR-frames are calculated using spectral ($F_{\mathrm{SPEC}}$) features with fps = 320 frames per second and $\#_{\mathrm{bins}} = 40$ bins (see Sections 4.1 and 4.2.2 for a more detailed account).

The autocorrelation curves are now truncated to lags $\tau \in \{12, \cdots, 80\}$, corresponding to element repetition frequencies from about 26.7 Hz down to 4 Hz. Similar to the Savi's Warbler detector, the motivation for this action lies in the high possibility of superior autocorrelation entries being located at small lags. Different from the former case, the small lags correspond to high frequencies. The high values at the mentioned positions are caused by their proximity to the global zero-lag maximum. Especially with signals containing lots of low frequency repetitions, the corresponding autocorrelation curves feature extensive slopes surrounding each maximum. Regarding the study of the Chaffinch song's structure, discussed in Section 2.4.2, the chosen frequency range appears to be sufficient for the majority of the expected stanzas. Thus, a first period estimate is drawn from the truncated autocorrelation curves by means of picking the respective global maxima:

$$\tilde{\rho}(n) := \underset{12 \leq \tau \leq 80}{\arg\max}(F_{\text{ACORR}}^{\text{norm}}(\tau, n)), \tag{5.19}$$

where the integer $n \leq L$ indicates the actual ACORR-frame position. $L$ refers to the total length of the extracted autocorrelation sequence. Actually, the above procedure is very similar to the extraction of the element repetition frequency in the Savi's Warbler detector. Note that the frequencies usually met in Chaffinch songs are quite low (2.6-25 Hz). As the frequency resolution of the $F_{\text{NPS}}$ features is quite low at the mentioned positions, the repetition period measure, being calculated by means of the above formula, serves as more precise indicator of such repetition rates. Contrary to the detection algorithm for the Savi's Warbler's song, in this procedure, false period detections constitute a critical factor. Usually, the phrases contained in the song of *Fringilla coelebs* are quite short while containing few elements. Hence, only few period estimates can be extracted from each phrase, which influences the overall detection performance. For this reason, an intermediate processing step is performed, improving the steadiness of the period estimates $\tilde{\rho}(n)$.

Sequentially processing the above estimates, the following Algorithm 2 estimates an expected period $\bar{\rho}_n$ for each position $n$, using a median of 5 of the previous outputs. Now, the actual period estimate is compared to the expected period. In the recordings of Chaffinch stanzas examined for this work, the inter-phrase shifts of the period, although being quite noticeable, do rarely reach twice of the element interspace. Thus, a period $\tilde{\rho}(n)$ is examined, if the estimate doubles the expected value. In this case, several local, "competing" maxima, are extracted from the associated autocorrelation curve $F_{\text{ACORR}}^{\text{norm}}(\cdot, n)$. Now, the competitor having the greatest possibility of being related to the expected period is chosen. Here, the local maximum's position is compared to the positions of multiples of the expected period. In this, a threshold is set to eliminate candidates being highly unrelated to the actual period estimate $\tilde{\rho}(n)$. Finally, comparing the autocorrelation energy of the best candidate and the estimate, a decision is made whether to replace the estimate by the new candidate:

Actually, the above algorithm is applied in a backward-forward manner. This has proven useful, because, as the end of the stanza-body is known, the above algorithm, reversely analysing the extracted periods, is likely to start with estimated period values containing valid information about actually present periodics. As, at this point, as there is no knowledge about the actual start or existence of a Chaffinch's stanza contained in the analysed stanza-body candidate, the period correction routine is more robust when run from backwards. As we want to correct both bisections and doublings of the period estimates $\tilde{\rho}(n)$, the repeated

**Algorithm 2**: Corrects pitch estimates by eventually reconsidering local autocorrelation maxima. Here, temporal information is incorporated to achieve a more stable pitch sequence.

```
Input: $\tilde{\rho}((L-1)-n)$, $F^{\text{norm}}_{\text{ACORR}}(\cdot, n)$, for $n \in \{0, \cdots, L-1\}$ ;     /* inversed period
    estimate */
Output: $\rho(n)$, for $n \in \{0, \cdots, L-1\}$
$\forall 0 \leq n \leq 5 : \rho(n) = \tilde{\rho}(n)$ ;                              /* copy first periods */
$\delta = 2$;
$n = 6$;
while $n \leq L$ do
    pitch_corrected = 0;
    $\bar{\rho}_n = \text{median}\{\tilde{\rho}(n-6), \cdots, \tilde{\rho}(n-1)\}$ ;                     /* reference period */
    if $\tilde{\rho}(n) > (2\bar{\rho}_n - \delta)$ then                                 /* check for jumps */
        $C = \{c_i \mid F^{\text{norm}}_{\text{ACORR}}(c_i, n) \text{ is local maximum in } F^{\text{norm}}_{\text{ACORR}}(\cdot, n) \wedge c_i \geq 12\}$ ;
        forall $c_i \in C$ do ;                             /* check multiples of reference */

            $m(i) = \underset{m \in \{1, \cdots, 5\}}{\arg\min} \, (|\bar{\rho}_n \cdot m - c_i|)$ ;          /* supposed harmonic's number */
            $e(i) = \frac{F^{\text{norm}}_{\text{ACORR}}(\bar{\rho}_n, n) - F^{\text{norm}}_{\text{ACORR}}(c_i, n)}{F^{\text{norm}}_{\text{ACORR}}(\bar{\rho}_n, n)}$ ;                        /* associated energy */
        end
        $D = \{(c_i, e(i)) \mid |\bar{\rho}_n \cdot m(i) - c_i| < m(i) \cdot \delta\}$ ;          /* plausible harmonics */
        if $(\underset{(c,e) \in D}{\max}(e)) > 0.3)$ then ;                        /* sufficient energy ?  */

            $(c_b, e_b) = \underset{(c,e) \in D}{\arg\max}(e)$ ;                      /* choose best competitor */
            $\rho(n) = c_b$;
        end
        if not pitch_corrected then
            $\rho(n) = \tilde{\rho}(n)$ ;                                    /* copy old pitch */
        end
    end
    $n++$;
end
```

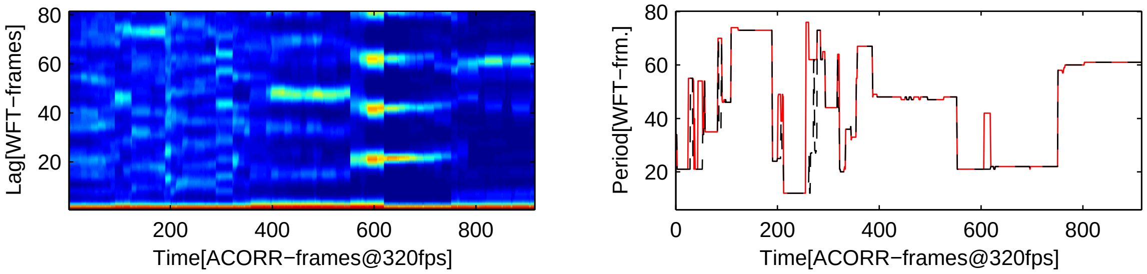


Figure 5.21: Left: autocorrelation feature sequence ($F_{\mathrm{ABP}}$) for a stanza-body candidate. Right: period estimate in red and corrected period in dashed black. The outlying pitches at frame 610 have been corrected.

application of the above algorithm, this time in a forward-directed manner, completes the task.

**Segmentation** In order to generate a preliminary segmentation, the corrected period curve $\rho(n)$ is now searched for areas of volatile period sequences. Inside these areas, an unsteadiness condition $c_1$ has to be fulfilled, defined as

$$c_1(n) := \begin{cases} 1 & \text{if } \sum_{k=n}^{n+2} |\rho(n+1) - \rho(n)| > 5 \\ 0 & \text{otherwise.} \end{cases} \tag{5.20}$$

Thus, the segments are chosen to fill out the regions lying in between the above unsteady areas. For each pair of such areas, a segment $s_i \in S^0$ is defined as the intercostal region having maximal length, being represented by the pair of starting and ending position $(p_i, q_i)$.

$$S^0 := \{(p,q) \mid (\forall p \leq n \leq q :\ c_1(n) = 0) \wedge (\nexists (v,w) \in S,\ v \leq p, q \leq w :\ w - v > q - p)\} \tag{5.21}$$

For each segment $s_i \in S^0$ a period histogram $H^\rho_{s_i}$ is generated, and the segment's dominant period $\rho(s_i)$ is extracted.

$$H^\rho_{s_i}(\tau) := |\{k \mid (\rho(k) = \tau) \wedge (p_i \leq k \leq q_i)\}|, \quad \text{for } s_i = (p_i, q_i), \tag{5.22}$$

$$\rho(s_i) := \arg\max_\tau (H^\rho_{s_i}(\tau)). \tag{5.23}$$

Now, the segment set $S^0$ is filtered from segments containing too many different periods, such as for example, a slowly increasing period.

$$S^1 := \left\{ s_i \in S^0 \;\middle|\; \left( \frac{\sum_{\tau=\rho(s_i)-2}^{\rho(s_i)+2} H^\rho_{s_i}(\tau)}{\sum_{\tau=12}^{80} H^\rho_{s_i}(\tau)} \right) > 0.85 \right\} \tag{5.24}$$

As there are some Chaffinch individuals featuring a continuous decrease of the element repetition period during large parts of their song, a second criterion is used to expand the actual segment set. At first, for each frame, a measurement is made examining the maximal period difference in a neighbourhood of 18.7 milliseconds. In this context, let

$$c_2(n) := \max_{1 \leq j \leq 6} \{\rho(n+j) - \rho(n)\}. \tag{5.25}$$

Similar to the above procedure, a histogram of the $c_2(n)$ measure is then used to extract the common range of short-time period deviations. Thereby, a narrow distribution of the deviation values is required: 90 percent of the deviations have to remain in the near neighbourhood of the most frequent deviation $d(s_i)$. The latter is required to be zero or negative, thus corresponding to a constant or increasing frequency. For $\delta \in \{-5, \cdots, 5\}$ and the segment $s_i = (p_i, q_i) \in S^0$, the above measures are given by

$$H_{s_i}^{c2}(\delta) := |\{k \mid (p_i \leq k \leq q_i) \wedge (c_2(k) = \delta)\}|, \quad \text{and} \tag{5.26}$$

$$d(s_i) := \underset{-5 \leq \delta \leq 5}{\arg\max}(H_i^{c2}(\delta)). \tag{5.27}$$

The segments satisfying the above conditions are added to the existing segment set.

$$S^2 := S^1 \cup \left\{ s_i \in S^0 \mid (-2 \leq d(s_i) \leq 0) \wedge \left( \frac{\sum_{\delta=d(s_i)-1}^{d(s_i)+1} H_{s_i}^{c2}(\delta)}{\sum_{\delta=-5}^{5} H_{s_i}^{c2}(\delta)} \right) > 0.9 \right\} \tag{5.28}$$

Reconsidering the strategy used for the primary segmentation, some areas may have been splitted erratically, being influenced by errors in the period estimation. Therefore, in a final segment postprocessing step, close (distance $<$ 42 ms) consecutive segments featuring an equal period are joined. Thereby, the resulting combined segment is required to comply with the histogram requirement met by the elements of $S^1$ (5.24). This time, no further effort is made to handle the segments featuring continuous period variations, leaving those segments untouched. In fact, the segments mentioned above are, by their nature, quite long. For the following classification procedure, a certain degree of segmentation is assumed and necessary. Thus, the previously mentioned segments are left disjointed. Hence, a cleaned, final segmentation of the stanza-body candidate extract is derived and saved in $S^3$. Let the segments $s_i \in S^2$ be numbered in an order ascending with the starting time $p_i$, thus $\forall s_i = (p_i, q_i) : p_i < p_{i+1}$ holds. Then, candidates for joint segments are gathered as denoted in Algorithm 3.

**Classification** Although some knowledge about the Chaffinch's song has already been incorporated into the segmentation procedure, in particular during the specification of valid periods and segment interspace durations, the following classification algorithm will use the knowledge obtained from the study described in Section 2.4.2. Based on this knowledge, a common stanza model is derived, describing a set of parameters to be measured in a valid stanza. In particular, for the detected periodic segments, their *number*, *duration*, *interspace* and *period* will be of importance. Furthermore, only 1-2 non-periodic gaps will be accepted.

Similar to the previous procedure, the approach described in the next paragraphs will explore the segmented extract starting from the "known" and advancing towards the "unknown", as depicted in Figure 5.23. Thus, as the stanza tail is presumed to be detected, the associated

**Algorithm 3**: Joins similar phrase candidates segments being close to each other.

```
Input: $S^2$;                                                   /* segment set */
Output: $S^3$ ;                                          /* joined segment set */
$S^3 = \{\}$; $k = 0$;
while $k < \big(|S^2| - 2\big)$ do
    $s_{tmp} = (p_k, q_{k+1})$ ;                    /* temporary joined segment */
    if $p_k - q_{k+1} < 14$ then                    /* check segment interspace */
        if $\left(\frac{\sum_{\tau=\rho(s_{tmp})-2}^{\rho(s_{tmp})+2} H^{\rho}_{s_{tmp}}(\tau)}{\sum_{\tau=12}^{80} H^{\rho}_{s_{tmp}}(\tau)}\right) > 0.85$ then
            $S^3 := S^3 \cup (p_k, q_{k+1})$ ;              /* add joined segment */
            $k = k + 2$;
    else
        $S^3 := S^3 \cup (p_k, q_k)$ ;                      /* add single segment */
        $k + +$;
end
return $S^3$
```

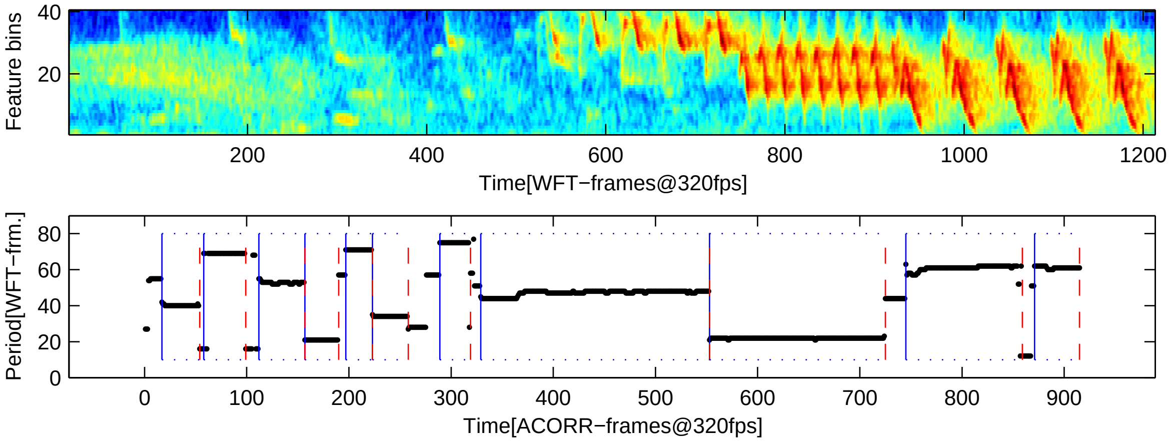


Figure 5.22: Final phrase segmentation of a stanza-body candidate. Top: spectral features of the excerpt. Bottom: dotted corrected period curve with boxes (blue line: start, red line: end), depicting the extracted phrase candidates.

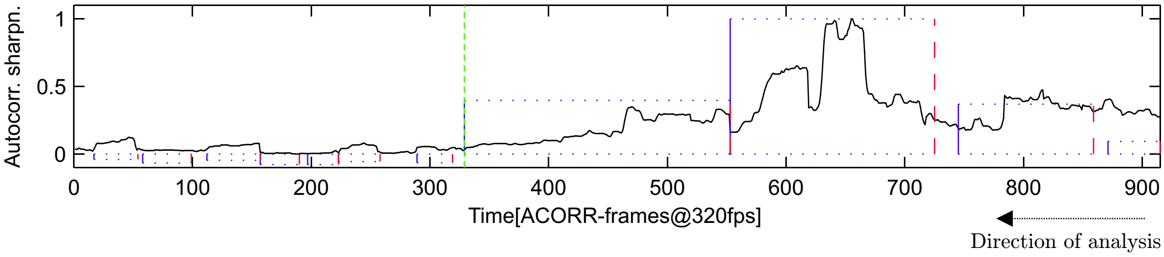


Figure 5.23: Continued example from Figure 5.22: autocorrelation sharpness measure and scored segments. The boxes' heights depict the individual scores. A green line marks the assumed start of the Chaffinch stanza.

stanza-body is processed in a reverse manner, starting at the flourish's beginning. Let $L + 1 = 900$ denote the (hypothetical) starting position of the stanza-tail, measured in ACORR-frames. This corresponds to a position of about 3.8 seconds, measured in relation to the extracted excerpt. At first, it is assured that a periodic segment exists ending in the range of 3.3 to 3.8 seconds (750-900). Given this case, the segments ending within 2 seconds before the stanza-tail are examined in the reversed order of their occurrence. For each such segment $s_i = (p_i, q_i)$, a score value $c_3(s_i)$ is computed indicating the deviation from an anticipated autocorrelation sum:

$$c_3(s_i) := \left( \sum_{k=p_i}^{q_i} F_{\text{ACSHARP}}(k) \right) - \bar{l} \cdot \text{median}\{F_{\text{ACSHARP}}(j) \mid 0 \leq j \leq L-1\} \tag{5.29}$$

Here, $\bar{l} = \frac{\text{fps}}{7.7}$ refers to a minimal segment length typically expected in Chaffinch songs. Note that as the ACORR-frames are computed each WFT-frame, the framerate of the former frames is identical to the WFT-fps value, leading to a minimum segment length of about an eighth of a second. Segments featuring a negative corresponding score value, thus containing relatively negligible periods, are sorted out at this point. In Figure 5.23, the previous segmentation is attached to its scores. Here, the noise-induced small segments in the stanza-body candidate have negative score values. As the following algorithm will try to trace a Chaffinch's stanza by means of testing the above segments in a reverse order, a segment set $S^4$ is defined:

$$S^4 := \{s_i \in S^3 \mid (q_i > 2\text{fps}) \wedge (c_3(s_i) > 0)\}, \tag{5.30}$$

for $s_i = (p_i, q_i)$. Let the elements of the above set be numbered in a reverse way, thus, for two segment ending positions $p_i, p_j$, $q_i > q_j$ for $i < j$ is required. While stepping from segment to segment in reverse succession, the inter-segment distance is measured. If the former distance exceeds a value of 50 milliseconds, a non-periodic segment is assumed in the stanza. As the number of such segments is restricted to a single one, the analysis is stopped at the second break exceeding the above length threshold. Now, the last segment being accepted by the algorithm is assumed to resemble the first stanza segment.

**Algorithm 4**: Traces the segments of the stanza-body candidate. Only a limited number of severe gaps is allowed. The algorithm also estimates the stanza starting time.

```
Input: $S^4$;                                          /* Ordered segment set */
Output: $k_s$ ;                       /* index of assumed first stanza segment */
$\#_{\text{badsegs}} = 0$ ;                    /* counts non-periodic segments */
$k = 0$;
while $k < \big(|S^4| - 2\big)$ do
    if $p_k - q_{k+1} < \frac{\text{fps}}{20}$ then          /* check segment interspace */
        $k++$;
    else if $\left(p_k - q_{k+1} < \frac{\text{fps}}{400}\right) \wedge (\#_{\text{badsegs}} < 1)$ then
        $\#_{\text{badsegs}}++$ ;                     /* increase non-period counter */
        $k++$;
    else break;
end
$k_s = k$;
return $k_s$
```

Thus, all stanza-body segments are expected to form the final segment set

$$S^5 := \{s_i \in S^4 \mid i \le k_s\}. \tag{5.31}$$

Now, a set of three criteria is used in order to perform a final classification of the prospected stanza-body. The first criterion, measuring the amount of accepted, periodic sequences detected in the excerpt determined by the above algorithm, is defined as follows:

$$c_4(n) \;:=\; \begin{cases} 1 & \text{if}\, l > 0.75 \cdot \text{fps} \\ 0 & \text{otherwise,} \end{cases} \tag{5.32}$$

$$\text{where} \quad l \;=\; \sum_{k=0}^{|S^5|-1} q_k - p_k, \quad (p_k, q_k) \in S^5. \tag{5.33}$$

An accumulated value corresponding to 750 ms of periodic phrases is considered to be sufficient. Now, the pitches contained in the selected segments are tested on their distribution. Using the knowledge drawn from the study discussed in Chapter 2.4.2, the following rules are applied: the set $S^5$ has to contain segments featuring periodics below as well as above 10 Hz.

$$c_5(n) := \begin{cases} 1, & \text{if}\, \exists s_i, s_j \in S^5 : (\rho(s_i) > 32), \wedge(\rho(s_j) < 32) \\ 0, & \text{otherwise.} \end{cases} \tag{5.34}$$

A last criterion ensures a minimum diversity on the segment's pitches.

$$c_6(n) := \begin{cases} 1, & \text{if } \exists s_i, s_j \in S^5 : |\rho(s_i) - \rho(s_j)| > 12, \\ 0, & \text{otherwise.} \end{cases} \tag{5.35}$$

Finally, the initial analysis excerpt corresponding to a stanza-tail candidate is classified as Chaffinch stanza-body if and only if $c_4 \wedge c_5 \wedge c_6 = 1$. A ranking based on the DTW-distance of the flourish is applied on the excerpts classified as Chaffinch. Alternatively, the sum of the segment scores $\sum_{s_i \in S^5} c_3(s_i)$ is used.

As for the precedent flourish-candidates, a SyncPlayer segmentation file is delivered as an optional output of this algorithm. The resulting representation is displayed in Figure 5.24. Using the Chaffinch flourish candidates of Figure 5.10, the algorithm has correctly detected 10 of 14 reasonable candidates. The remaining candidates were false positives. Furthermore, an acoustic representation of the results is generated as described for the audio summarisation algorithm (Sec.5.2.3).

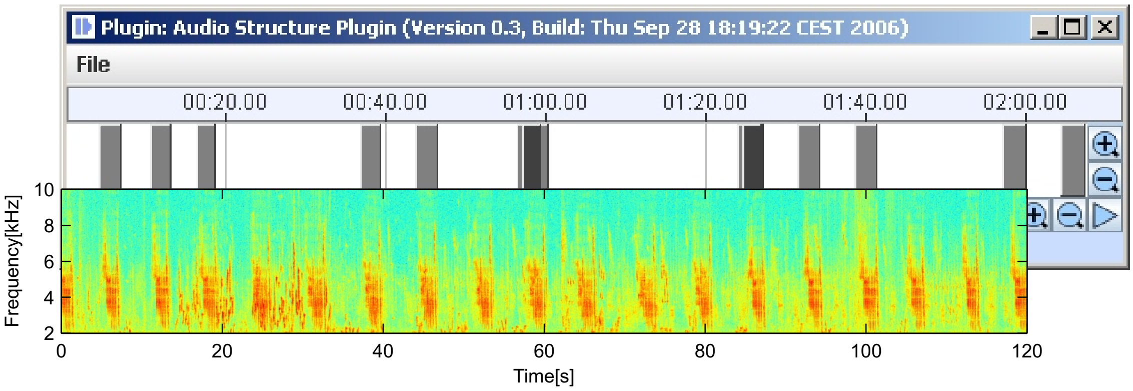


Figure 5.24: Continued example from the flourish detection section (Fig. 5.10, p. 73). SyncPlayer representation of the final Chaffinch's stanza detection results.

### 5.4.2 Autocorrelation vector clustering

In an early stage of the development of the above segmentation and detection routines, some experiments were performed, exploring the feasibility of the application of speech recognition paradigms on bird voices. Both forms of acoustic information propagation feature a highly structured nature: in this section, the Chaffinch's stanza will be considered as similar to a sentence in human speech. Here, the equivalent of a word or syllable will be the phrase and element, respectively, in birdsongs. Now, the experiment was made to model a Chaffinch's stanza by means of training an hidden Markov model (HMM), being widely used to capture the (hierarchic) structure of human speech, using autocorrelation features of the birdsong stanzas.

Thus, building models reflecting the periodic structure of the Chaffinch's song, the intention of keeping the aspired model invariant to the single individual's proprietary element shapes, as depicted in spectrograms, was followed. Although, due to the harsh inter-individual differences in the stanza structures (see Fig. 2.6, p.19), a general stanza model could not be derived (see following section), the segmentation methods used as a preprocessing step to the model estimation, turned out to be highly performant. Moreover, the training procedure used for the HMM, now incorporating similarities of several stanzas, provides an improvement on the existing segmentation. Possessing the potential to improve the segmentation performed in the previous chapter as well as in the future research, the approach followed in the stated experiment will be lined out in the following paragraphs. Basically, the training procedure of a stanza model consists in the following steps:

1. *Extract* $F_{\text{ABP}}$ features of several Chaffinch stanzas.
2. *Reduce* the feature dimension.
3. Perform a $k$-means *clustering* on the whole data set, reflecting the phrase structure of the birdsong stanzas.
4. *Train* a hidden Markov model based on the segmentation induced by the above classification.

The autocorrelation features used to represent the Chaffinch stanzas are extracted using spectral features similar to those utilised in the previous chapter. Analysing the 2-7 kHz frequency band, 150 of the 40-bin features are extracted each second. The resulting $F_{\text{ABP}}$ features, containing a maximum lag of 59 samples, are extracted with the same sampling rate, thus $\text{fps}_{\text{ac}} = 150$. Now, as explained in Section 4.2.5, the autocorrelation curves are downsampled by a factor of 5, reducing the number of autocorrelation coefficients to 12. The size of the training data set is directly affected by the framerate. Thus, a high framerate, resulting in about 400 data points per stanza, is particularly important for the success of the whole procedure.

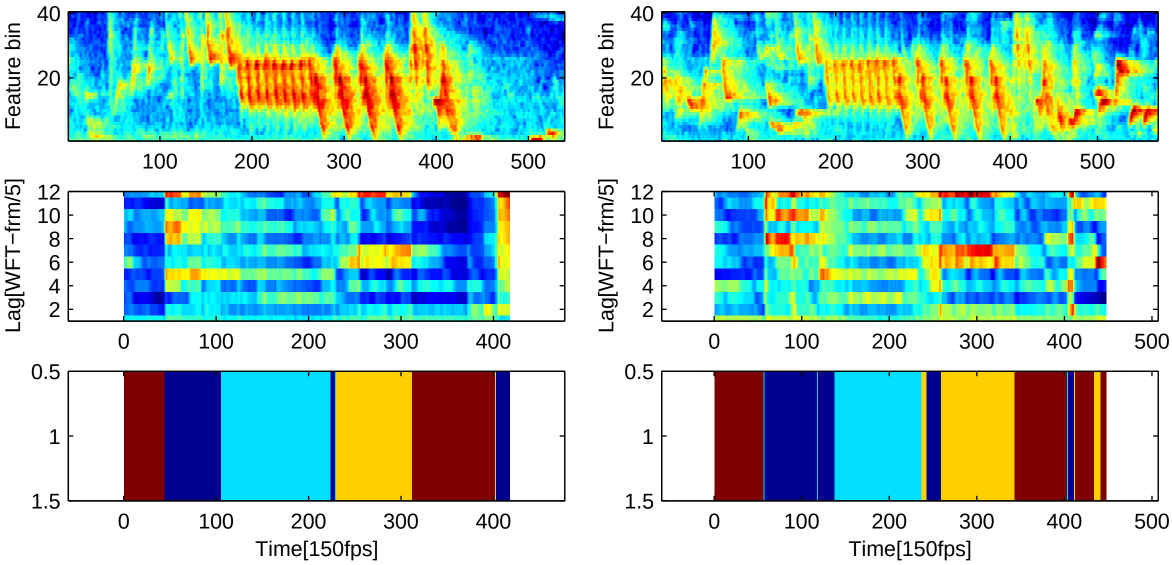


Figure 5.25: Result of $k$-means clustering for two of three Chaffinch stanzas. Top: spectral features. Mid: low-resolution $F_{\text{ABP}}$ features. Bottom: clusters in colour, corresponding to numbers (left to right): brown (4), blue (1), light blue (2), orange (3).

Having extracted the set of training data represented by several series of normalized autocorrelation curves, the whole data set is clustered using the *k-means* algorithm. In general, the goal of a clustering algorithm is to define a set of $k$ clusters by means of partitioning a set of feature vectors. Thus, each feature vector in the data set is associated to a cluster. Thereby, a distance measure, indicating the summed distance between the data vectors and the centres of their associated clusters, is minimised. Considering the $k$-means implementation from the NETLAB toolbox, implementing the techniques described in [Nab02], the squared Euclidean distance of two autocorrelation curves is used: Let $X := (x_1, \cdots, x_M) \in \mathbb{R}^{12 \times M}$ be the sequence constructed through the concatenation of all $F_{\text{ABP}}$ sequences. Given a cluster centroid $c^j$, the distance measure $d : \mathbb{R}^{12} \times \mathbb{R}^{12} \to \mathbb{R}$ is defined as

$$d(x_i, c^j) := \sum_{\tau=0}^{11} (x_i(\tau) - c^j(\tau))^2 \tag{5.36}$$

For a fixed number of clusters, e.g. $k = 5$, the centroids $c_0^i$, $i \in \{0, \cdots, k-1\}$ are initialized using values randomly drawn from the set of the training data. By iteratively calculating the new cluster assignments of the features and the cluster centroids, the following iteration converges, minimising the overall cluster to cluster centroid distance. Note that the described process might also converge to local minimums of the distance function. In our experiments, the iteration was stopped after 100 cycles.

**Algorithm 5**: Basic k-means clustering

```
   Input: $F_{\text{ABP}}$ sequence $X$
   Output: Clusters $C_{N-1}^i$
1  for i=0 to k-1 do
2      choose $r \in \{0, \cdots, M-1\}^{12}$ randomly;
3      $c_0^i = \left(x_{r(0)}(0), \cdots, x_{r(11)}(11)\right)$ ;              /* initialize centroids */
4  end
5  for n=0 to N-1 do
6      for i=0 to k-1 do
7          $C_n^i = \left\{ j \mid i = \underset{i \in \{0,\cdots,k-1\}}{\arg\min} \left(d(x_j, c_n^i)\right) \right\}$ ;   /* actualize clusters */
8      end
9      for i=0 to k-1 do
10         $c_{n+1}^i = \frac{\sum_{i \in C_n^i} X_i}{\|\sum_{i \in C_n^i} X_i\|}$ ;   /* compute new centroids */
11     end
12 end
```

Note that the sum used to calculate the new cluster centroids in line 10 of Algorithm 5 denotes the component wise sum of the respective vectors, resulting in the cumulative vector. Fixing the number of clusters to little more than the expected number of phrases turns out to be suitable for data sets containing the stanzas of a single individual. When operating with data from multiple individuals, the number of clusters required, unfortunately, seems to increase linearly with to the number of individuals.

A hidden Markov model is now initialized and trained using the data and labels from the $k$-means classificator. For the experiments performed in this work, the implementations of the “Hidden Markov Model (HMM) Toolbox”, written by Kevin Murphy [Mur], were used. Being fairly often used to model the transitions of phonemes, syllables and other sequential entities in human speech (as exemplified in [Rab89]), these models appear well-suited to represent the phrase structure inherent to chaffinch songs. The bottom part of Figure 5.25, depicting automatically assigned cluster numbers, may be interpreted as a state sequence of a process, modelling a singing Chaffinch: at first, there is mostly non-periodic noise (state 4). Then, the Chaffinch sings a first phrase (state 1), which is followed by the singing of two further phrases (state 2 and 3), ending with a rather non-periodic flourish (again state 4). Using the language of regular expressions, both of the depicted stanzas resemble the form $4^* \, 1^+ \, 2^+ \, 3^+ \, 4^*$. This example could be easily refined by adding further states, but, here, the generalisation caused by the small set of states turns out to be quite helpful. In particular, state number 4 represents a whole class of aperiodic signals, including the motif of the Chaffinch’s flourish. Although differing in their spectral representations, the $F_{\mathrm{ABP}}$ features of aperiodic signals are quite similar to each other. Now, the sequence of the “bird’s states” is modelled as a statistic process, representing the bird’s state as discrete random variable $q : \mathbb{N} \to \{1, \cdots, 4\}$, which may be evaluated for each ACORR-frame. On the left side of Figure 5.26, this “stanza process” is modelled in a simplified manner. Thus, we may ask for the probability of the bird singing phrase number 3 after singing the sequence 1 1 1 2 2. This is a typical question to be solved by a model of a statistical process. For the special case of a Markov model, the mentioned probability only depends on the previous process state. Considering the given example, this is “phrase 2”. Thus, for a Markov model, given the example state set $S = \{s_i \mid 1 \le i \le 4\}$,

$$P\left(q(t) = s_i \mid q(t-1) = s_k, q(t-2) = s_j, \cdots\right) \; = \; P\left(q(t) = s_i \mid q(t-1) = s_k\right) \tag{5.37}$$

holds. As, assuming a Markov model, the absolute time position does not influence the above transition, a matrix $S$, containing the probabilities of all possible state transitions, can be defined by

$$S_{i,j} := P\left(q(t) = s_j \mid q(t-1) = s_i\right), \quad \text{for } 1 \le i, j \le 4. \tag{5.38}$$

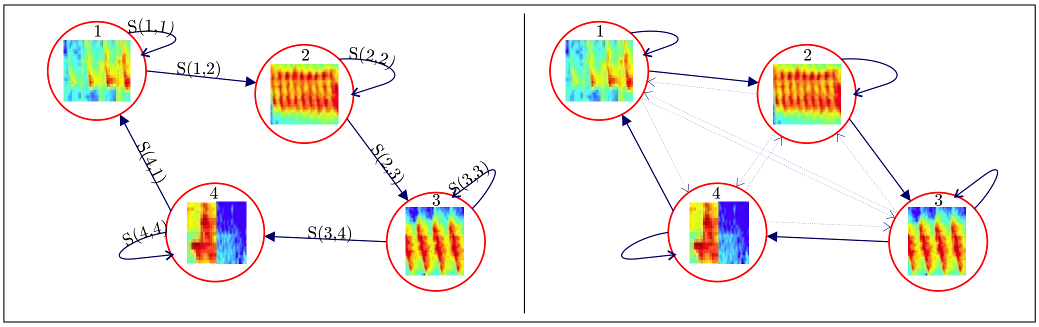


Figure 5.26: Two HMM’s modelling a phrase sequence. Left: simple circular model. Right: implemented model with low possibilities for off-circle state transitions. Instead of the $F_{\mathrm{ABP}}$ features used for the actual computations, spectral features are used for the thumbnails.

In Figure 5.26, some of these transition probabilities are notated on top of the state-transistions. The right HMM depicts the full range of possible state transitions. The matrix $S$ is stochastic, thus fulfilling the conditions $S_{i,j} \ge 0$ and $\sum_{j=1}^{4} S_{i,j} = 1$ for all $i, j \in \{1, \cdots 4\}$. In order to calculate the probability of a full sequence, the additional knowledge on a starting distribution $R_i := P(q(0) = s_i)$ is needed. Now, assuming the individual state transitions to be independent, the probability of a phrase sequence $W = (w_0, \cdots, w_k)$, being sung by the bird, is calculated as follows:

$$P(W) = P\left(q(0) = w_0, \cdots, q(k) = w_k\right) \;=\; R_{w_0} \prod_{n=1}^{k} S_{wn-1,wn} \tag{5.39}$$

Unfortunately, the bird's singing state sequence is not known given the acoustic recordings analysed in this work. Rather, we are limited to only knowing the acoustic output of the bird. Actually, we are dealing with the periodicity features of it, being derived from noisy and distorted recordings. In fact, we are assuming an invisible, hidden model, while observing the acoustic features. Concerning the hidden Markov model, each state $s_i$ has an associated function $P(o(t) \mid q(t) = s_i)$, measuring the probability of a distinct observation $o(t)$ being made with the model residing in the respective state. As it is the case in this section, the observations are represented by the single autocorrelation curves $F_{\mathrm{ABP}}(\cdot, t)$. Thus, the observed autocorrelation curves are connected to the assumed singing states, using the indirection of the above probability function. Now, the probability of a $F_{\mathrm{ABP}}$ sequence, for $V = (o(1), \cdots, o(T))$, given our envisaged model, can be calculated by means of summation over the probabilities of all $r^{\max} = |S|^T$ possible hidden state transition sequences $W^r$ having length $T$,

$$P(V) = \sum_{r=1}^{r^{\max}} P(V \mid W^r) P(W^r). \tag{5.40}$$

This is efficiently achieved using the Forward Algorithm, which has the complexity $O(|S|^2 T)$, and is explained in detail in [DHS00]. As for the probability of a phrase sequence (equation 5.39), the probability of a sequence of observations is derived by calculating the product of the associated probabilities $P(o(t) \mid w^r(t))$.

For the models derived in this work, the observation-state probabilities mentioned above are defined using multivariate Gaussian density distributions. Given the clusters $C^i_{N-1}$, $i \in \{1, \cdots 4\}$, derived in the previous $k$-means algorithm, such a distribution is initially defined by calculating the mean vector $\mu_i = \mathrm{mean}(C^i_{N-1}) \in \mathbb{R}^{12}$, as well as the covariance matrix $\Sigma_i \in \mathbb{R}^{12\times 12}$, for each of the four clusters. Now, the multivariate Gaussian density function

$$p_i(o(t)) \;:=\; \frac{1}{(2\pi)^{d/2} \mathrm{det}(\Sigma_i)^{1/2}} \cdot \exp\left[-\frac{1}{2}(o(t) - \mu_i)^\top \Sigma_i^{-1} (o(t) - \mu_i)\right], \tag{5.41}$$

approximating $P(o(t) \mid s_i)$, is associated to each state $s_i$. Here, $d = 12$ expresses the feature dimension, and $\Sigma_i^{-1}$ refers to the inverse of the respective covariance matrix. Applying this model on the $F_{\mathrm{ABP}}$ sequences, representing the birdsong stanzas, the $k$-means classification result should be reflected by the respective probability outputs of the hidden states' distributions.

Approaching the HMM training process, the state transition matrix $S$ and the initial distribution $R$ are initialised using random values. After ensuring them to fulfil the stochastic conditions mentioned above, the variables $R$ and $S$, as well as the parametrisations of the

Gaussian models $\Sigma_i$, $\mu_i$, are bundled to form the HMM's parameter set $\theta$. Maximizing the HMM's probability for the given data by modification of $\theta$, the iterative Baum-Welch algorithm is used to estimate the values in $S$ and $R$, regarding the available stanza sequences as training data. Furthermore, the algorithm is allowed to modify the parameters of the Gaussian models, adapting them for more uniform state transitions. Thus, the HMM is trained to provide high likelihoods concerning the stanzas' features given as example observation sequences. Thereby, the knowledge on the individual stanza instantiations is used to build a more general model.

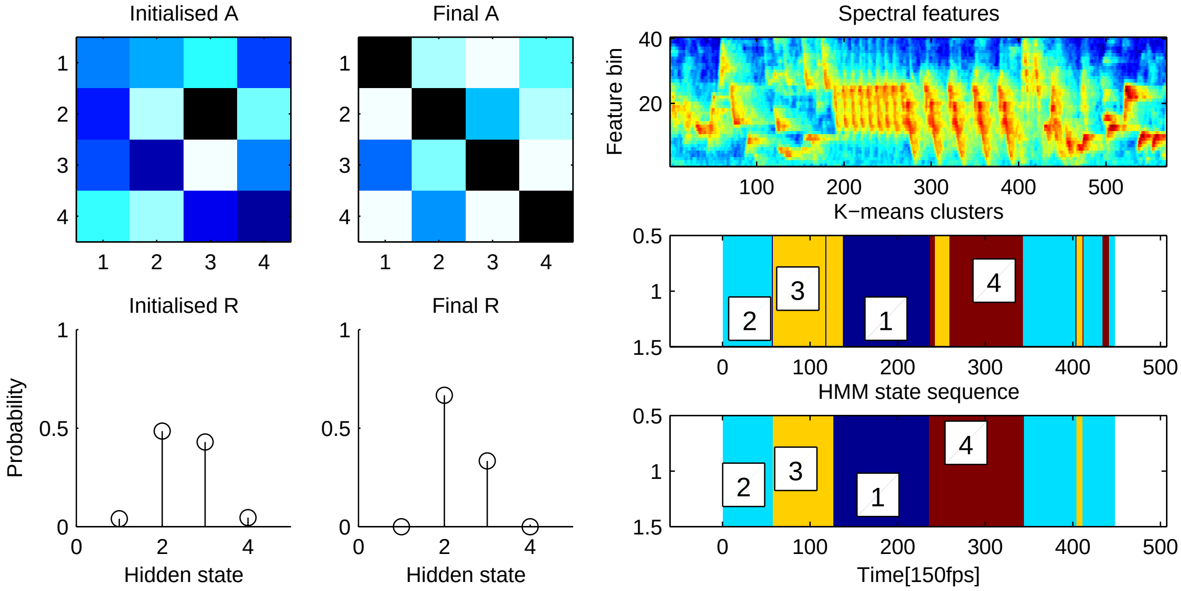


Figure 5.27: Left column: HMM configuration resulting from the Baum-Welch algorithm, compared to the initial conditions. Dark colours in the above (logarithmically scaled) matrices depict high transition probabilities. Right column: spectral features, $k$-means clusters and most likely state sequence of a Chaffinch's stanza in colour, corresponding to numbers (left to right): light blue (2), orange (3), blue (1), brown (4).

In the left column of Figure 5.27, the state transition matrix $S$ and prior $R$, are depicted both before and after the training process. Continuing the example given in Figure 5.25, the $F_{\mathrm{ABP}}$ autocorrelation sequences of three Chaffinch stanzas were used as data for the training, supporting each state's Gaussian model ($\Sigma_i$, $\mu_i$) with about 300 data points. Considering the final state transition probabilities, the diagonal elements of $S$, being associated with intra-state transitions, reflect the stationary form of the phrases' autocorrelation features. Furthermore, considering the bottom image, displaying a supposed hidden state sequence fitting the actual features, the lighter coloured fields of $S$ reveal their meaning: in every row, there is a second coloured maximum, pointing to the column of the state which is likely to be followed by the state associated to the row. For example, state (row) 3 is likely to be followed by state one. This can be verified by the state sequence. Here, state 2 represents the more or less non-periodic sequences, framing the actual stanza. Plotted in the center row of Figure 5.27, the final starting probability $R$ reflects the fact of two of the three stanza examples starting at state 2, while a single stanza is estimated to start at state 3.

Actually, deriving the most probable hidden state sequence, as shown in the previous example, is a non-trivial task, as, at this point, there are several means of defining optimality. Here, the Viterbi algorithm, using dynamic programming techniques, successes in extracting the path displayed in Figure 5.27. For further explanation on this and other standard algorithms used with HMM's, we refer to [Rab89].

As exemplified in the previously delineated experiment, the application of clustering methods such as $k$-means and the time-encoding hidden Markov model on the robust periodicity features, derived in this work, state an promising tool for the segmentation of birdsongs featuring a periodic structure. The repetition of almost identical stanzas, as performed by Chaffinch individuals, is an essential prerequisite to the introduced approach, although a more general model, describing several stanza types, would be realisable.

Initially, the above HMM training procedure was designed as part of a Chaffinch detector: A composite HMM was build, joining the previous stanza model and a "residual model". The latter model was trained using a wide range of audio sequences containing typical background noise, also featuring various birdcalls different from the Chaffinch's song. In the experiments performed with a variety of such composite models, it turned out that the individuals' stanza structures did not permit the generalisation needed for a generic Chaffinch detector. However, the identification of a single individual's stanzas worked quite well. Here, the data sets used for training and tests were disjunct.

# Chapter 6

# Evaluation

## 6.1 Detection of Savi's Warbler's song

The Savi's Warbler detector, being described in Section 5.3.1, has been evaluated on a data set containing 19 hours of 4-channel monitoring recordings from Lake Parstein in Brandenburg, Germany as part of a research and development project founded by BfN and BMU. As part of this project, having the founding identifier 806 82 060 - K2, several publications dealing with the detection of Savi's Warbler were released or are in preparation ([CIB07] [BWC08], [ASA08]). A microphone array, autonomously recording the sounds of the protected area, has been set up by our cooperative colleagues Karl Frommolt and Klaus-Henry Tauchert at the Animal Sound Archive, Humboldt University Berlin. The sensors were installed on a boat anchored in the center of the mentioned Lake, where they were left alone for most of the operation time. As this project was conducted from April to June, the recordings, being timed to take place during the sunrise and sunset periods, contain a wide range of different birdsongs as well as a mixture of diverse background noises. Envisaging the evaluation of a robust detector, the set has been chosen to uniformly cover the whole range of background noise met at the particular monitoring site. Furthermore, the signal excerpts were divided into 4 background noise classes, separating the signals featuring heavy wind or rain noise from those with many birdsongs as well as quiet recordings from particularly clear Warbler recordings.

Most of the monitoring excerpts used in the following study have a total length of about 15 minutes, although the length of a few examples deviates in a range of about 5 minutes. For each of the excerpts, occurrences of the Savi's Warbler's song were manually annotated. The annotation of such a large data set turns out be quite time consuming: the complete 19 hours were listened to by means of a four-channel (2 times stereo) high fidelity equipment. Here, a slight equalising of the audio signal was applied, suppressing the low frequencies while sustaining the frequency range containing the Warbler's song. Furthermore, the overall speaker volume was adapted to the signal's amplitude, thus amplifying quiet excerpts. Additionally, a mixed-down real-time spectrogram of the actual signal was displayed and examined by the human annotator. Using an annotation tool specially developed for this application, the perceived Warbler songs were journalised, listing the occupied time interval as well as a perceptiveness estimate. The estimate was determined by the listener, and rated on a scale of 5 classes. In this, the first class corresponds to a "barely perceivable" song, followed by the conditions usually met, which include simultaneous birdsongs and wind noise, being assigned class values of 2 or 3. The further classes correspond to superiorly perceivable songs.

In this section, the results of the given detection algorithm are compared to the annotations performed by the human listener. Of course, several conditions affecting the direct comparison of the two annotation procedures have to be noted. First, the discovery of a song made by the observer is not logged instantaneously. A latency of about 1 up to 3 seconds, partly depending on the loudness of the song in relation to the surrounding noise, has to be considered. Besides this, the reaction time of the observer, being subject to various influences, has not been precisely estimated. For the detection algorithm, the actual implementation discards song candidates being shorter than three seconds. However, the usual timing accuracy of the algorithm's output is estimated to a second. Thus, for the annotations of short, call-like Warbler sounds, there is a low probability for the automatic and manual annotation to match each other. Consequently, annotations referring to excerpts shorter than 5 seconds are discarded from the evaluation. Furthermore, a tolerance area of 4 seconds was added at the beginning and end of each manually annotated segment. Actually, as shown in Table 6.1, the influence of the mentioned short excerpts is quite low. In fact, considering the manual annotations, the discarded excerpts constitute a fraction of 5 percent of the overall singing time.

| **Category** | **Complete data** | **Friendly** | **Manybirds** | **Quiet** | **RainWind** |
|---|---|---|---|---|---|
| **Data** | 19h 1min. | 4h 15min. | 4h 29min. | 5h 27min. | 4h 30min. |
| **Manual** | 12h 28min. | 4h 9min. | 4h 23min. | 2h 29min. | 1h 26min. |
| **Manual5+** | 11h 52min. | 4h 8min. | 4h 17min. | 2h 13min. | 1h 13min. |
| % **Det. OK** | 92.95% | 99.59% | 97.28% | 79.06% | 80.55% |
| % **Det. ?** | 1.22% | 0.71% | 1.12% | 1.42% | 1.60% |

Table 6.1: Summary of the Savi's Warbler detector's evaluation, by signal categories. The first two rows label the absolute, "**Manual**ly" annotated Warbler singing time and the time associated to stanzas more than 5 seconds long ("**Manual5+**"). **"Det. OK"** contains the recall values: the percentage of (5+)-song time annotated manually also being found by the detector. **"Det. ?"** measures the percentage of possibly falsely detected time spans.

As depicted in the above comprehensive table, the Warbler's song is widely present (66% of the whole timespan) in the data set used for this study. Thus, considering the algorithmic detection, false positives are more "difficult to achieve". In fact, about 40 percent of the segments counted in the "**Det. ?**"-row of this table actually represent Savi's Warbler songs which the human listener failed to identify. Considering the raw statistics, not incorporating the short call-segments and tolerance area mentioned above, the proposed algorithm still detects 88.4% of the manually annotated singing time. Although an evaluation of the number of stanzas being actually detected would be appreciable, the task of annotating such excerpts, while dealing with multiple overlapping Warbler songs, turns out to be challenging for the human observer. For the algorithmic aspect, this task, requiring the integration of acoustic source seperation techniques, is left for future work.

### 6.1.1 Signal quality categories

In the following section, the signal categories mentioned above will be analysed in detail. The influences of each signal type on the detection rate will be depicted, accompanied by a spectrogram and the frequency distributions of representative excerpts. As the signal levels were not altered in between the individual recordings, an absolute volume can be measured for each frequency coefficient. Note, that all of the following examples contain a more or less obvious Savi's Warblers song. Consequently, the frequency distributions on the right of the spectrograms have a common energy peak close to 4 kHz.

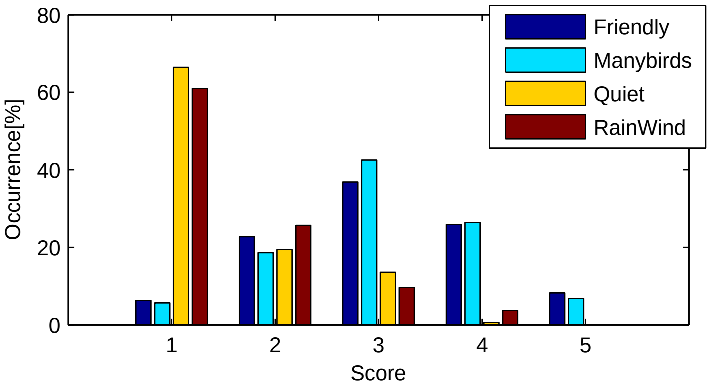


Figure 6.1: Distribution of the signal classes with regard to the perceptiveness scores. The Quiet and RainWind classes share a common, low score for different reasons.

In Figure 6.1, the manually annotated perceptiveness score is individually evaluated for each signal class. For the detection statistics of the individual monitoring excerpts, ordered by signal class, see page 108.

**Friendly** There are a few occasions when the Savi's Warbler's song can be listened to with only little influence of any interferences. Such outstanding recordings are usually performed during the day or a few hours after sunset. At these times, many of the other species are no longer active. The Warbler is usually found singing in the reed near to the microphones, and very few wind or rain is heard if at all. As Figure 6.1 shows, most of the annotated calls are easy to identify for the human annotator. Likewise, the detection rate of the algorithmic detector reaches the best performance. The example given in Figure 6.2 shows a song at -23 dB, given the score 4.

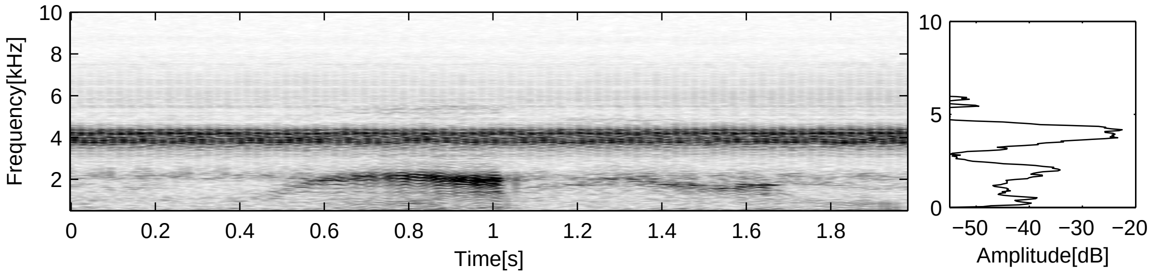


Figure 6.2: Left: spectrogram of a recording featuring a near Savi's Warbler's song. Right: mean absolute energy distribution. Recorded on 04.26.2007, at 23:14.

**Manybirds** Recorded at the times of major interest for bird watchers, namely dusk and dawn, these recordings contain a vast number of overlapping birdsongs and calls. In order to drown the songs of their competitors, the individuals usually elevate the intensity of their singing. Thus, the signal-to noise ratio, concerning the song of a Savi's Warbler, is ruled by the relative distance of the Warbler, followed by the intensity of its song. As the recordings of this category are likely to contain the songs of several Warblers singing simultaneously, the distribution of these sounds on the four recorded channels also constitutes an important influence. Although, for the human annotator, the song of the Savi's Warbler is easy to recognise, the overlapping birdsongs represent a challenge to the automatic annotator. With the used $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features being highly robust and invariant to the aperiodic mixture of the surrounding birdsongs, the Warbler's period is still well represented. Moreover, the remaining periodics of the birdsong background come with a slow repetition frequency, thus being easy to distinguish from the targeted 50 Hz frequency. The song depicted in Figure 6.3 was given the score 2.

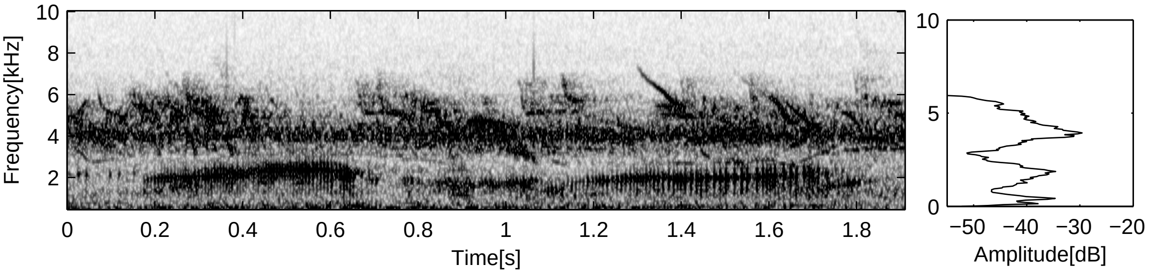


Figure 6.3: Left: spectrogram of a "full" recording. Right: mean absolute energy distribution. Recorded on 05.07.2007, at 4:59.

**Quiet** The quiet recordings, although featuring a low absolute noise level, are characterized by a relatively shy, if any singing behaviour of the avian inhabitants at Lake Parstein. As some occasional noises, mostly caused by wind, remote airplanes or trains dominate the signal's volume, the signal-to noise ratio is rather low for a distant Savi's Warbler's song. Given the steady stream of hiss which is perceptible at higher amplification levels, the long and constant song of a distant bird is easy to miss. Although the applied features are likely to yield clear periodicity parameters even from these songs, the autocorrelation sharpness threshold criterion (see equation 5.11), designed to discard questionable candidates, prohibits the detection of such vague events. This situation, exemplified in Figure 6.4, is mostly found in recordings performed at late night. The warbler song depicted here arrives with a decreased volume of -61 dB, thus having 38 dB less energy than the example for the "Friendly" class. A score value of 1 was given to the actual song.

**RainWind** When recording under bad weather conditions, the signal-to noise ratio of the target signal is likely to be low. Especially when the microphones are subject to serious air turbulences, caused by heavy wind, the recorded data is heavily distorted up to a complete loss of any relevant signal. As the actual version of the detection algorithm is not capable of combining the separated song excerpts, many candidates do not reach the sufficient length for

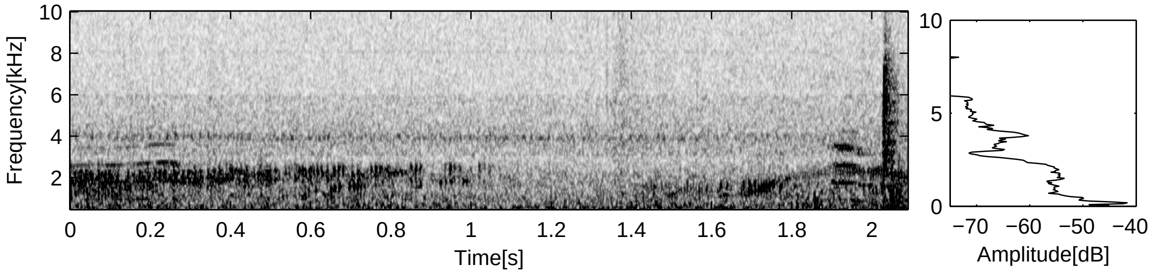


Figure 6.4: Left: spectrogram of a quiet recording. Right: mean absolute energy distribution. Recorded on 05.03.2007, at 23:14. Note the overall -20 dB drop in energy.

a positive classification. Figure 6.5 displays such a fragmented recording. "Rainy" recordings are padded with a constant noise floor caused by the rain. This happens to the extend of some bird voices being completely masked by the sound of raindrops. In the special case of the recordings considered for this evaluation, a plastic canvas, although protecting the microphones from the water, tends to amplify the recorded rain noise. With the masking caused by heavy gusts of wind, this effect is mainly held responsible for the degraded detection performance. Moreover, given a loud ambience, most of the birds living around the monitoring area, including the Savi's Warbler, tend to decrease the overall rate of their songs.

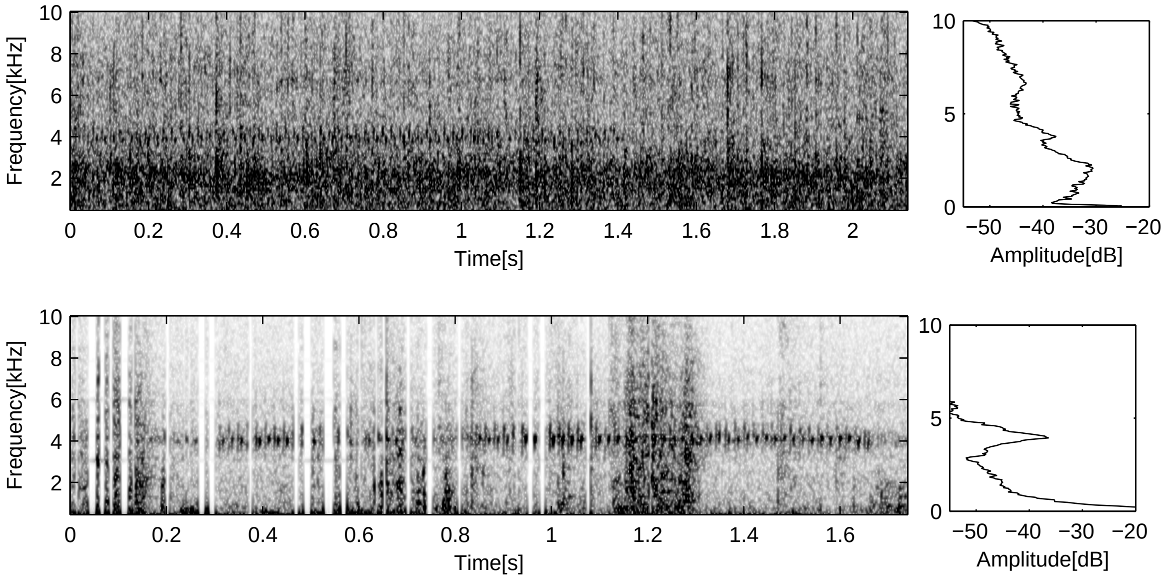
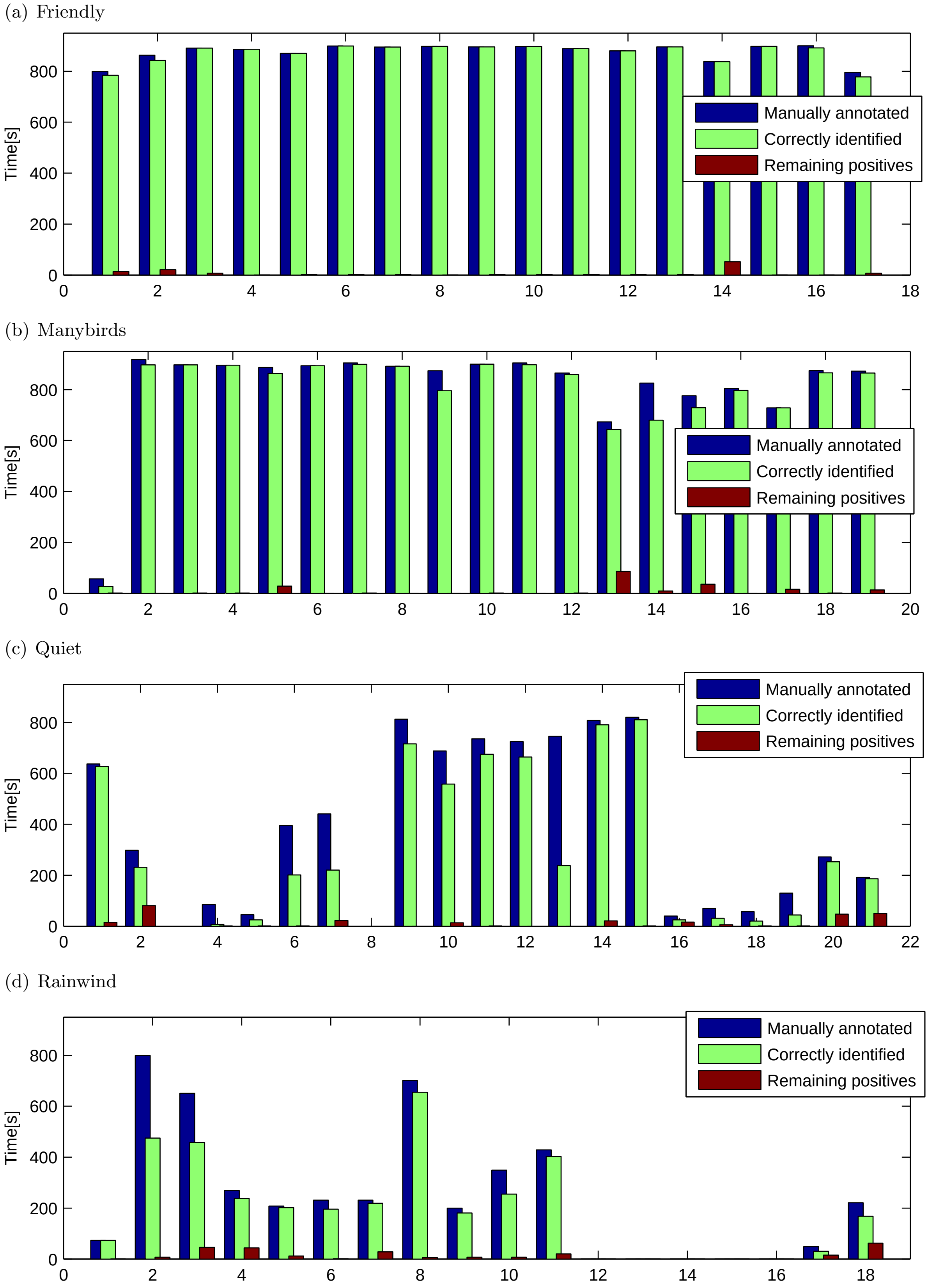


Figure 6.5: Left: spectrogram and of a rainy (top) and windy (bottom) recording. Right: mean absolute energy distribution. Note the signal drop-outs probably caused by an overload of the A-D converter.



(a) Friendly

(b) Manybirds

(c) Quiet

(d) Rainwind

Figure 6.6: Recognition results for individual monitoring recordings. Blue bars represent the manually annotated Savi's Warblers singing time. Green bars count the seconds being annotated by both the manual and automatic detector. Brown bars display the remaining automatic annotations.

### 6.1.2 Performance

The Savi's Warbler detector was implemented using the MATLAB® scripting language, following the goal to design an off-line detection routine while benefiting from the numerous tools available in this framework. A typical 4-channel, 48000Hz, 16bit PCM audio file of 15 minutes length is analysed in about 16 minutes, measured on an AMD AthlonXP 2800+ with 1024 MB RAM. Considering modern computer hardware, the analysis can actually be performed in less than the time recorded. Here, the direct application of the second analysis step on the whole test signal is considered, omitting the energy-based segment preselection routine. As the recordings are split into several analysis excerpts to be processed separately, the actual processing time grows linearly with the length of the input signal.

In this main part of the analysis, most (95%) of the CPU time is used for the $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ feature extraction step. Here, the initial WFT used for the extraction of high-resolution spectral features consumes 70% of the overall time. A filter bank approach might be pursued instead to achieve a more efficient extraction of frequency-band limited features. Secondly, the calculation of the short-time autocorrelations and the subsequent fast Fourier transforms take a dominant (20%) part. Adjusting the tradeoff between computational costs and annotation precision, the ACORR-frame rate $\mathrm{fps}_{\mathrm{ac}}$ constitutes an ideal parameter to adapt the algorithm's performance. Although a smaller number of ACORR-frames decreases the number of features finally gathered in the time binning step, the autocorrelation costs, as well as the costs of the subsequent steps decrease in direct proportion to this frame rate. Considering this as an adjustment of the algorithm's robustness up to temporal occlusions of the Warbler's song, two similar approaches can be applied on the number of the subbands used for the feature extraction as well as the number of microphones and channels. While the former of these parameters enables the control of the algorithms frequency selectivity, a large number of channels increases the chance of targeted birds being directly focused by a microphone. Furthermore, as discussed in Section 4.2.6, the computation of the $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features may be sped up by circumventing the computation of autocorrelation measures. In this case, a replacement has to be found for the autocorrelation sharpness measure.

Considering the preselection routine, the actual implementation uses a filtering approach to extract the energy curve of a signal. Thus, for the above machine and 15 minute-records, the preselection routine is completed in less than 2 minutes, including the time used for reading the data from hard disc. As the coverage of the following analysis is determined by the actual signal, no prediction will be made on the subsequent computational costs. However, the above numbers regarding an $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ analysis of the complete signal determine an upper limit of these costs.

## 6.2 Detecting the Chaffinch's song

The generic Chaffinch stanza detector, as described in Sections 5.2.2 and 5.4.1, like the previous detector, has been implemented in the MATLAB® scripting language. Both of the mentioned detectors are part of a bioacoustic signal processing tool kit, providing an easy-to-use interface to the developed feature extractors.

The following Table 6.2 contains some evaluation results for two disjunct data sets. The first set was compiled from monitoring recordings containing some Chaffinch calls. The recordings were performed at 3 different places, featuring different noise conditions. The second set was gathered from a greater number of monitoring recordings, containing a diverse set of bird sounds and noise. No Chaffinch stanzas were found when manually inspecting this set of recordings.

| Set | Compl. data | #Stanzas | #Flourish cand. | #Det. stanzas | #False det. |
|---|---|---|---|---|---|
| 1 | 19.7min. | 91 | 226 | 44 | 15 |
| 2 | 59.5min. | 0 | 669 | 0 | 30 |

Table 6.2: Evaluation results for the Chaffinch detector.

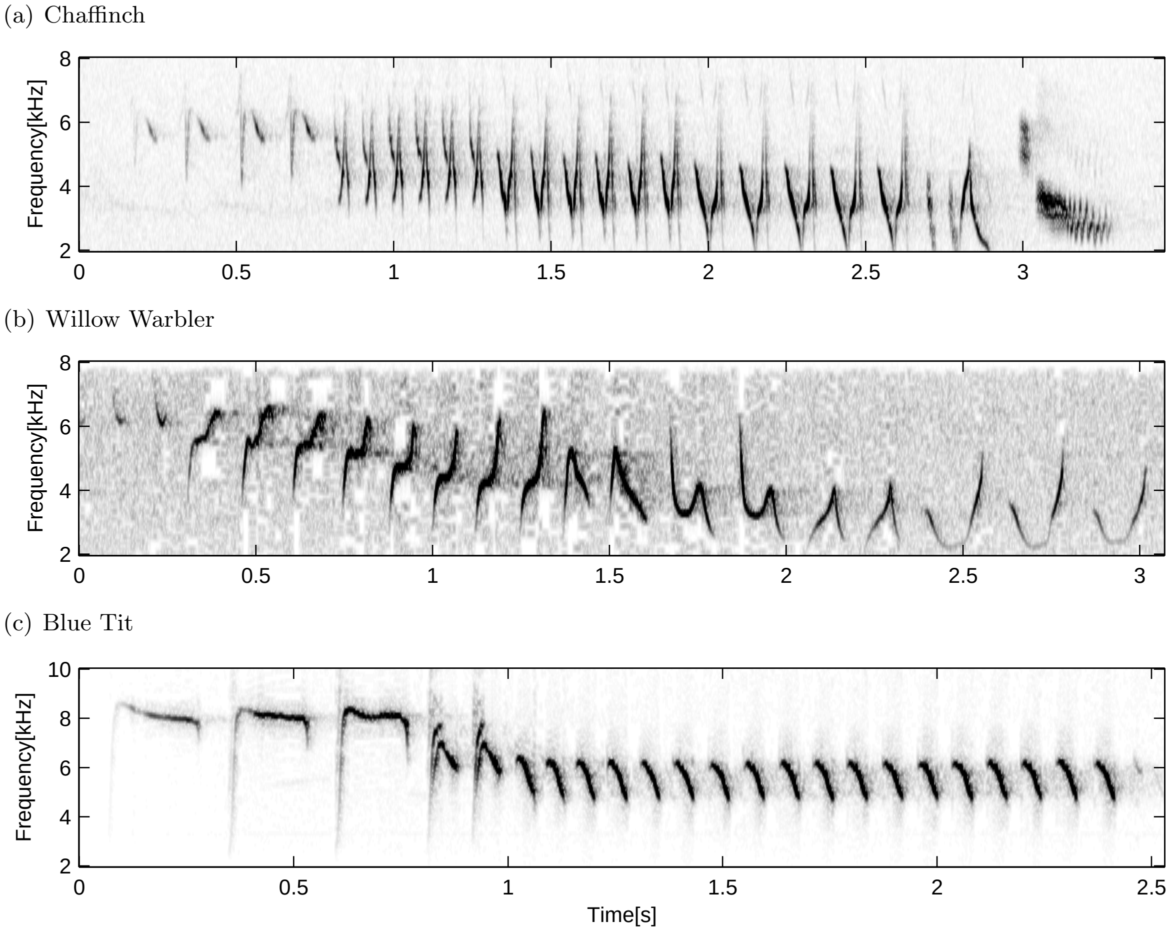
As depicted in this table, the overall detection rate nearly reaches 50%. Moreover, there are lots of false detections, amounting to 25% of all detections for the first test set. Still, the final detection stage filters out a lot of false candidates. This is clearly visible when taking into account that a nearly constant amount of about 220 candidates is proposed by the flourish detection stage for each 20 minutes. Considering the ratio of candidates and false positives in both test sets, only 5.29% of all false candidates were also falsely identified as Chaffinches in the model-comparison step. As there are way too much flourish candidates being extracted from the second test set, improving the candidate extraction routine promises to significantly enhance the precision of the proposed algorithm.

With this perspective in mind, a further evaluation was performed on the first of the data sets covered by the above table: the flourish candidates were precisely determined by manual annotation. Afterwards, the model-based Chaffinch detector was applied on the chosen Chaffinch songs in order to determine the false-negative rate of this second step of the Chaffinch detector. Unfortunately, only 41 (45%) of the 91 calls were correctly identified. The loss of classification precision can be explained by the smaller number of flourish candidates. For the case of several flourish candidates being associated to the same stanza, the model is fitted at slightly varying positions within this stanza. This allows for a more variable application of the model, resulting in an increase of detections.

As the above results imply, the Chaffinch detector itself has not grown past an experimental stadium: as the inter-individual diversity of sung stanzas' is quite large even for this species, the generalised model used to detect the stanza-body comes with a high rate of false positives. Most of the false detections are represented by a mixture calls being performed simultaneously by a couple of birds. As these mixtures often contain a multiplicity of periodic signals, the generic Chaffinch model is likely to fit some of them. This is because the model of the Chaffinch's stanza structure is quite general, permitting a large set of phrase combinations. The individual phrases of different birds may now occur in a combination fitting the model of a Chaffinch stanza. There are also a couple of other birds, like the Willow Warbler (*Phylloscopus trochilus*) or the Blue Tit (*Cyanistes caeruleus*), which feature a similar stanza structure. Their stanzas are depicted in Figure 6.7. However, many people also tend to mistake the singing Willow Warbler for a slowly singing Chaffinch.

Moreover, the preselection stage, searching for the Chaffinch's flourish, adds to this phenomenon. As the number of flourish candidates to be extracted from a certain time span is fixed, a high number of false candidates is generated in the absence of Chaffinch songs. Given a few Chaffinch songs, the false detections may be sorted out by utilisation of the DTW-ranking. Here, the correct matches are likely to achieve score values superior to the scores of false candidates.

In order to evaluate further weak points of the algorithm to be improved in future work, the


(a) Chaffinch

(b) Willow Warbler

(c) Blue Tit

Figure 6.7: Spectrograms from stanzas of a Chaffinch (a), Willow Warbler (b) and Blue Tit (c). The latter birds' songs are often falsely detected as Chaffinch stanzas by the Chaffinch detector.

detector's decisions were logged in the second, semi-automatic part of the study. Here, four criteria responsible for the acceptance of a Chaffinch candidate were defined according to the classification criteria introduced in Section 5.4.1.

- *High energy near flourish*: the quick check, as depicted in Figure 5.20, has to return a positive result when performed on the energy progression of the stanza candidate.

- *Final periodic segment*: a periodic segment, constituting a phrase candidate, has to be detected near the flourish candidate (see Algorithm 4).

- *Stanza length*: Equation 5.32 requires a minimum time span being covered by phrase candidates with sufficient autocorrelation sharpness values.

- *Period diversity*: as defined in Equation 5.34, the phrase candidates have to include element repetition periods below and above 10 Hz. According to Equation 5.35, a minimum period range has to be maintained in the remaining candidates.

| Energy | Final seg. | Length | Diversity |
|---|---|---|---|
| 9% | 0% | 27% | 12% |

Table 6.3: Chaffinch detector: detection criteria and their responsibility for discarding actually correct Chaffinch stanzas.

As depicted in Table 6.3, the segment extraction and stanza tracing routines are to be held responsible for the majority of false negatives. Here, improvements could be made especially in the pitch correction and segment scoring steps. As a working pitch correction routine is fundamental for the following segment extraction, a more adaptive routine may lead to extensive improvements in detection precision. When calculating the segments' individual scores in Equation 5.29, the median of the autocorrelation sharpness of the full candidate excerpt is a more or less weak reference score. Here, a more global reference could increase the reliability of the filtering of weak segments previous to the segment combination step.

# Chapter 7

# Conclusions and Future Work

Focussing on the computational interpretation of the vast data sets being accumulated in monitoring sessions, some basic signal features and detection tools are developed in the thesis on hand. Here, special attention is given to the particular requirements of the discussed realistic monitoring scenario, including low signal-to noise ratios and unpredictable signal interferences. As a main contribution of this work, two sets of robust features and their extraction methods are introduced. All of the features are designed to measure acoustic parameters of significance for birdsongs, but each feature set turns out to be suitable for a particular group of applications. The first feature set, namely the spectral features, encodes a rough sketch of the spectral parameters of a monitoring signal. Besides serving as a basis for some fast candidate extraction routines, they form the representations used in the proposed template matching routines. The second feature set robustly extracts the parameters of periodic repetitions of elements within the spectrogram. For some species which perform their songs using a particular, species-specific repetition frequency, robust detectors are build using the periodicity features. The following paragraphs subsume the mentioned feature sets, their applications and potential.

**The spectral features** facilitate the application of template matching methods. Here, a Dynamic Time Warping approach is used to implement an audio summarisation routine, which may be used to condense the material to be presented to a human investigator or an algorithmic classifier. For this summarisation task, no prior knowledge on the avian inhabitants of the monitoring area is needed.

Contrarily, the detection of a particular species seems to require more individual feature parametrisations and classification routines. So, the template matching approach used in the Chaffinch detector, searching for a typical flourish, is based on a mannerism of this particular species. Considering the low-level spectral features, including the energy and spectral flatness measures, the provided information is satisfactory for a preselection of candidates, enabling a speedup of the whole detection procedure.

**The periodicity features,** measuring parameters of periodic element repetitions, turn out to be suitable when searching for birdsongs featuring periodic sequences. When applied in such a scenario, the periodicity features show a superior robustness up to the usual background noise found in field recordings: as birds often vocalise a very precisely timed series of elements, their songs can be easily distinguished from noise using these features. In the Savi's Warbler detector, the denoised novelty autocorrelation ($F_{\mathrm{NPS}}^{-\mathrm{nois}}$) features are used to robustly detect the typical element repetition frequency used by this species. The mentioned features constitute the most robust features developed in this work and the average detection rate of the above detector is reasonably high (92.95%). The $F_{\mathrm{ABP}}$ and $F_{\mathrm{NPS}}^{-\mathrm{nois}}$ features may be easily adapted to serve the description of other species' sounds, including crickets and frogs. Lowering the frequency resolution of the underlying spectral features, an application to the detection of birds like the Great Tit or the Eurasian Bittern becomes feasible. For the latter bird, a similar approach is described by Bardeli [Bar08].

Actually, the identification and extraction of parameters being representative for a distinct species generally is a great challenge. In the second stage of the Chaffinch detector, a complex structure extraction algorithm is used. Here the phrase structure is considered to represent the species' peculiarity. Adaptive band autocorrelation and autocorrelation sharpness features were used to robustly extract the parameters of the single phrases. Unfortunately, the generic nature of the model, as our evaluation shows, appears to lead to a high number of false positive matches.

A more general structure extraction algorithm was presented, using $k$-means clustering techniques. Moreover, periodicity based hidden Markov models were introduced, using the Chaffinch's stanza as an example. In future work, this approach may lead to more elaborate models for several bird species. Another step towards generic classification of birdsongs has been made by introducing the generic periodic features $F_{\mathrm{abstractp}}$. Here, Neuronal Nets, Support Vector Machines or HMM's may be trained in an unsupervised manner. The latter approach may be applied to a large set of animals that emit periodic acoustic signals. Hence, a variety of detectors may be derived from the data sets existing in today's animal sound archives.

As it has been stated in the introductory sections, some of the previously recapitulated approaches remain in an experimental stadium. Considering the juvenile status of the research branch concerned in computational bioacoustics, this is not astonishing. Here, the goal was to investigate basic methods facilitating several topic-related tasks. Although some of these methods, being generated as by-products of the detection algorithms, have not been evaluated on a big scale, their experimental results encourage further research on the underlying concepts.

# Bibliography

[ADM96] S. E. Anderson, A. S. Dave, and D. Margoliash. Template-based automatic recognition of birdsong syllables from continuous recordings. *Acoustical Society of America Journal*, 100:1209–1219, August 1996.

[ASA08] SFA ASA, EAA. Acoustics'08. Paris, 2008.

[AUD] AUDACITY. Free audio editor and recorder. Website. `http://audacity.sourceforge.net/`.

[Bar08] Rolf Bardeli. *Algorithmic Analysis of Complex Audio Scenes*. PhD thesis, University of Bonn, 2008.

[BH82] Hans-Heiner Bergmann and Hans-Wolfgang Helb. *Stimmen der Vögel Europas*. BLV Verlagsgesellschaft, München, 1982.

[BML] BMLIST. Bioacoustic monitoring - communications of an international expert group on bioacoustic monitoring. Mailing list. `https://mailbox.iai.uni-bonn.de/mailman/listinfo.cgi/bioacoustic-monitoring`.

[BNF06] T. S. Brandes, P. Naskrecki, and H. K. Figueroa. Using image processing to detect and classify narrow-band cricket and frog calls. *Journal of the Acoustical Society of America*, 120:2950–2957, November 2006.

[BWC08] Rolf Bardeli, Daniel Wolff, and Michael Clausen. Bird song recognition in complex audio scenes. In *Computational bioacoustics for assessing bioacoustics*, number 234 in BfN-Skripten, 2008. `http://www.bfn.de/fileadmin/MDB/documents/service/skript234.pdf`.

[Che01] E. David Chesmore. Application of time domain signal coding and artificial neural networks to passive acoustical identification of animals. *Applied Acoustics*, December 2001.

[CIB07] University of Pavia CIBRA. Xxi ibac international bioacoustic congress. Pavia, 2007.

[CLRS01] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001.

[DHS00] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[Ewe07] S. Ewert. Effiziente Methoden zur hochauflösenden Musiksynchronisation. Diploma thesis, 2007.

[Fag04] S. Fagerlund. Automatic recognition of bird species by their sounds. Master's thesis, Helsinki University of Technology, 2004.

[FBC08] Karl-Heinz Frommolt, Rolf Bardeli, and Michael Clausen, editors. *Computational bioacoustics for assessing biodiversity. Proceedings of the International Expert meeting on IT-based detection of bioacoustical patterns*, 2008.

[Fre06] C. Fremerey. SyncPlayer - a Framework for Content-Based Music Navigation. Diploma thesis, 2006.

[FT07] K.-H. Frommolt and K.-H. Tauchert. Anwendungsszenarien eines auf bioakustischer Mustererkennung basierenden Monitorings. *Verhandlungen der Gesellschaft für Ökologie*, 37, 2007.

[IUC] IUCN. IUCN Red List of Threatened Species 2007. Website. http://www.iucnredlist.org.

[KM98] J. A. Kogan and D. Margoliash. Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden markov models: A comparative study. *Journal of the Acoustical Society of America*, 103(4):2185–2196, April 1998.

[Mat06] H. Mattes. Effiziente Synchronisation von Musikdatenströmen. Diploma thesis, 2006.

[MSHRD05] A. Mitschke, C. Sudfeld, H. Heidrich-Riske, and R. Dröschmeister. Das neue Brutvogelmonitoring in der Normallandschaft Deutschlands - Untersuchungsgebiete, Erfassungsmethode und erste Ergebnisse. *Vogelwelt*, 126, 2005.

[Mur] Kevin Murphy. Hidden markov model (hmm) toolbox for matlab. Website. http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html.

[Nab02] Ian Nabney. *NETLAB: algorithms for pattern recognition.* Springer-Verlag New York, Inc., New York, NY, USA, 2002.

[Pos02] G. Possart. Signal classification of bird voices using multiscale methods and neural networks. Master's thesis, University of Kaiserslautern, 2002.

[Rab89] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[RHH$^+$05] R.S. Rempel, K.A. Hobson, G. Holborn, S.L. Van Wilgenburg, and J. Elliott. Bioacoustic monitoring of forest songbirds: interpreter variability and effects of configuration and digital processing methods in the laboratory. *Journal of Field Ornithology*, 76, 2005.

[RJ93] L. Rabiner and B. Juang. *Fundamentals of speech recognition.* Prentice Hall, 1993.

[SAS] SASLAB. Avisoft-saslab pro. Website. http://www.avisoft.com.

[SDK$^+$03] F. Schwenker, C. Dietrich, H.A. Kestler, K. Riede, and G. Palm. Radial basis function neural networks and temporal fusion for the classification of bio acoustic time series. *Neurocomputing*, 51:265–275, 2003.

[STT07] Arja Selin, Jari Turunen, and Juha T. Tanttu. Wavelets in recognition of bird sounds. *EURASIP J. Appl. Signal Process.*, 2007(1):141–141, 2007.

[SYN] SYNC. Syncplayer homepage. Website. http://audentify.iai.uni-bonn.de/synchome/.

[TSA] TSAWeb. Homepage of the animal sound archive, berlin. Website. http://www.tierstimmenarchiv.de.

[VB88] B. Van Veen and K. Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Magazine*, 5:4–24, April 1988.

[XBA] XBAT. Extensible bioacoustic tool. Website. http://xbat.org.

# Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig angefertigt, keine anderen als die angegebenen Hilfsmittel benutzt, sowie Zitate kenntlich gemacht habe.

Bonn, den

Autor