Luberadzka *et al.*
*EURASIP Journal on Audio, Speech, and Music Processing* (2024) 2024:27
https://doi.org/10.1186/s13636-024-00350-w

EURASIP Journal on Audio, Speech, and Music Processing

EMPIRICAL RESEARCH

Open Access

# Towards multidimensional attentive voice tracking—estimating voice state from auditory glimpses with regression neural networks and Monte Carlo sampling

Joanna Luberadzka[1*], Hendrik Kayser[1], Jörg Lücke[1] and Volker Hohmann[1]

**Abstract**

Selective attention is a crucial ability of the auditory system. Computationally, following an auditory object can be illustrated as tracking its acoustic properties, e.g., pitch, timbre, or location in space. The difficulty is related to the fact that in a complex auditory scene, the information about the tracked object is not available in a clean form. The more cluttered the sound mixture, the more time and frequency regions where the object of interest is masked by other sound sources. How does the auditory system recognize and follow acoustic objects based on this fragmentary information? Numerous studies highlight the crucial role of top-down processing in this task. Having in mind both auditory modeling and signal processing applications, we investigated how computational methods with and without top-down processing deal with increasing sparsity of the auditory features in the task of estimating instantaneous voice states, defined as a combination of three parameters: fundamental frequency F0 and formant frequencies F1 and F2. We found that the benefit from top-down processing grows with increasing sparseness of the auditory data.

**Keywords** Computational auditory scene analysis, Voice state tracking, Auditory feature extraction, Neural networks

## 1 Introduction

Selective auditory attention is essential in most real-life acoustic environments. Human listeners without hearing impairment tune out the irrelevant acoustic clutter and attentively follow sound objects with ease, but from the machine listening perspective, selective attention is a challenging task. For example, despite many technological advances, hearing aids still tend to amplify the background sounds together with the signal of interest. Other than filtering the sound based on specific properties like the spectral range, direction of arrival, or degree of interaural correlation, hearing devices have no ability to follow a specific sound object. The goal of this study is to contribute to understanding what computational strategies of human audition are still missing in the audio algorithms.

Computationally, following an auditory object can be illustrated as tracking its acoustic properties, e.g., pitch, timbre, or location in space. Results of previous studies indicate that to track voices in a crowded acoustic space, a fusion of several dimensions representing different acoustic properties is required. These properties can be estimated based on the auditory features extracted from the acoustic signal. The difficulty is related to the fact that in a complex auditory scene, the information about the tracked object is not available in a clean form. The more cluttered the sound mixture, the more time and frequency regions where the object of interest is masked by other sound sources [1, 2]. Previous studies suggest that the remaining sparse time-frequency regions where

*Correspondence:
Joanna Luberadzka
joa.luberadzka@gmail.com
[1] Carl von Ossietzky Universitat, Oldenburg, Germany

the voice of interest dominates over other sound objects (*auditory glimpses*) are essential in decoding auditory scenes. Glimpses provide robust information about the voice of interest and, hence, can be used as reliable cues for tracking.

However, the incomplete *glimpsed* information taken out of the context may be ambiguous at times: sparse glimpses alone may not provide enough evidence to be linked with a unique possible underlying cause. Many scholars believe that solving this ill-posed problem is possible due to the *top-down processing* involved in perception [3]. In contrast to the feed-forward processing system in which the information travels straight from input to output in a direct way, the perception is frequently described as a top-down processing architecture, where the input features are confronted with the expectations formed at the higher level of abstraction. In a top-down system, the output depends both on the input and on some prior beliefs, which are a set of constraints restricting the possible outcomes of the task. By allocating the neural resources to the regions of expected high importance, the brain simplifies the task of making sense of fragmentary information available in a complex auditory scene.

In our previous work [4], we proposed a computational model of attentive tracking of competing voices [5], which combined the top-down and bottom-up processing. The model was used to track $F0$ of two simultaneous voices based on sparse periodicity-based auditory features (sPAF) [6–8] extracted from the mixture of voices. It was realized with sequential Monte Carlo sampling (particle filters) [9], coupled with simple analytically designed probabilistic $F0$-models (described in detail in [4]). This process simulated attentive tracking in humans. We found that although the information carried by sPAF extracted from the mixture of two voices is sufficient to simultaneously track both $F0$s, the knowledge of $F0$ alone is not sufficient to correctly segregate the features. Our results confirmed that more voice properties need to be estimated to solve the attentive tracking task.

In this study, we extend the previously used system as follows: (1) instead of tracking only $F0$, we track voice states consisting of three parameters (fundamental frequency $F0$ and formant frequencies $F1$ and $F2$); (2) in the feature extraction, we include energy-based features instead of using solely periodicity-based features; (3) instead of a likelihood model of F0 for the periodicity-based features, we propose a joint $F0$, $F1$, $F2$-likelihood model for combined periodicity- and energy-based features. $F0$, $F1$, and $F2$ are related to speech production and perception and provide critical cues for the identification of speech sounds. $F0$ corresponds to the rate of vibration of the vocal cords, which determines the perceived pitch of the sound. $F1$ and $F2$—the first and the second lowest resonant frequency in the vocal tract—are influenced primarily by the position of the tongue during speech production. They are both found to be critical for distinguishing between different vowels.

We investigate the potential of a new approach including all three parameters. Firstly, we test how much the performance is affected by the increasing sparsity of the auditory features. We bypass the segregation problem and let the model estimate the 3 dimensional state based on already segregated target-related sPAF. We generate continuously voiced signals with defined state trajectories, extract sPAF features, and simulate a varying degree of difficulty in the sPAF features. Secondly, to investigate the benefit of top-down processing, we compare the proposed model with two classes of methods without top-down processing: non-sequential Monte Carlo sampling and regression neural networks. In the first class, a straightforward Monte Carlo simulation is used [10]: competing hypotheses are distributed across the possible range of parameter values and evaluated with the same likelihood model as used in the particle. This can be understood as a particle filter without a continuity model. In the second class, the regression neural network [11] learns the mapping between the sPAF and the voice state and applies it to predict the most likely instantaneous voice states. We know from [12] that this approach is successful for estimating state of a single voice. However, it was not clear whether this purely bottom-up approach would be able to deal with sPAF extracted from a more difficult auditory scenes, where there are less target-related glimpses.

The main contributions of this study are:

1. Extension of the previously published voice state likelihood model (from F0 estimation to F0-F1-F2 estimation)
2. Comparison of the sampling-based voice tracking approach with a regression neural network
3. Introduction of a novel, perceptually motivated F0 tracking error measure

This paper is organized as follows: In Section 2, we introduce implementation details of the auditory feature extraction, four state estimation methods, and $F0$, $F1$, $F2$-data likelihood models. Section 3 reviews conditions in which the methods were evaluated and describes the performance measures. Section 4 guides the reader through the results, and Section 5 discusses the results in a broader context.

## 2 Methods

### 2.1 Sparse periodicity-based feature extraction

Feature extraction in this study is motivated by human auditory processing. We used the approach developed in [6–8] called *sparse periodicity-based auditory feature* (sPAF) extraction. We adopted this approach in [4] to track *F*0 of two competing voices. The method was designed to blindly extract auditory glimpses from the sound mixture. In particular, the auditory glimpses are here defined as salient tonal components across frequency. The sPAF extraction consists of three main stages:

1. Auditory pre-processing: This pre-processing stage simulates the sound processing in the peripheral auditory system, including:
   (a) Acoustic modification due to the middle ear implemented as a band-pass filter (0.5–2 kHz)
   (b) Spectral analysis in the cochlea implemented as filterbank of 23 gammatone band-pass filters
   (c) Dynamic range compression in the cochlea implemented as a power-low with an exponent of 0.4
   (d) Neural transduction from the vibrations of the basilar membrane in the cochlea to electrical stimulation of the auditory nerve implemented as a half-wave rectification, followed by the 5th-order lowpass-filter at 770 Hz, and 40 Hz high-pass filter.

   The auditory pre-processing stage yields 23 time-domain signals, which are forwarded to the periodicity analysis stage.
2. Periodicity analysis: In each frequency channel *c*, the periodicity analysis [13] is performed every 20 ms to reveal the dominant periods in the analyzed time instance of a signal. Around each considered time step *n*, eight signal segments of duration P′ are formed, as depicted in Fig. 1A.2. P′ is varied from 0.0014 s (1/700 Hz) to 0.0125 s (1/80 Hz). The eight signal segments are averaged, yielding a *base function* $v_{cn}(\mathrm{P}')$. The energy of the signal that spans all eight signal segments is termed *total energy* $E_{tot,cn}(\mathrm{P}')$. It is calculated as the mean square amplitude of that signal. The energy of the base function is termed *periodic energy* $E_{P,cn}(\mathrm{P}')$ and it is computed as the mean square amplitude of the base function. Lastly, for each point in time *n*, channel *c* and each tested period P′, the *normalized periodic energy* $\mathrm{synch}_{\mathrm{cn}}(\mathrm{P}')$ defined as the ratio of the periodic energy and the total energy is computed:

$$\mathrm{synch}_{\mathrm{cn}}(\mathrm{P}') = \frac{\mathrm{E}_{\mathrm{P,cn}}(\mathrm{P}')}{\mathrm{E}_{\mathrm{tot,cn}}(\mathrm{P}')}. \tag{1}$$

   Values of $\mathrm{synch}_{\mathrm{cn}}(\mathrm{P}')$ close to one indicate a high degree of tonality in a time-frequency bin *cn* and are treated as a footprint of speech in the sound mixture.
3. Glimpsing Values exceeding a certain threshold (e.g., $> 0.9$) indicate an auditory glimpse. Features tied to this dominant period characterize the glimpse. In [7, 8], a glimpse was defined by four quantities: period $P_{cnm}$, total energy $E_{cnm}$, interaural time difference $T_{cnm}$, and interaural level difference $L_{cnm}$, where the indices *c*, *n*, and *m* denote the frequency channel, time instance, and glimpse index within a channel, respectively. In our recent work, we used the period $P_{cnm}$ to track voice fundamental frequency. The period is a sufficient feature for F0 estimation, but to estimate a 3-dimensional voice state consisting of F0, F1, and F2, it is necessary to examine the energy as a function of frequency. Hence here, we use both period and total energy.

In summary, in every time instance *n* at the output of the glimpsed feature extraction stage, we obtain an observation $O(n)$:

$$O(n) = \{G_{cn} | c = 1, ..., 23\}, \tag{2}$$

which consists of 23 *glimpse sets* $G_{cn}$ for each channel *c*. $G_{cn}$ can consist of a single value, multiple values, or no value at all. The salient *glimpses* within the set $G_{cn}$ are denoted as $\vec{g}_{cnm}$:

$$G_{cn} = \{\vec{g}_{cnm} | m = 1, ..., M_{cn}\}, \tag{3}$$

where $M_{cn}$ is the number of elements in the set (number of glimpses in channel *c* at time *n*). Each element $\vec{g}_{cnm}$ consist of two values: *glimpse period* $P_{cnm}$ and *glimpse energy* $E_{cnm}$:

$$\vec{g}_{cnm} = [P_{cnm}, E_{cnm}]. \tag{4}$$

The number of salient glimpses in a set depends on the content of the analyzed signal. Glimpses obtained with this method in one instance of time can be visualized as glimpse patterns, shown in Fig. 1 (B). They represent all the salient periods found across 23 frequency channels and their corresponding total energy values. For more implementation details about the sPAF extraction, the reader is referred to [4, 6–8, 14].

#### 2.1.1 Observation vector for the neural network

The number of glimpses varies depending on the content of the acoustic signal: in general the more complex the auditory scene, the less salient glimpses available.

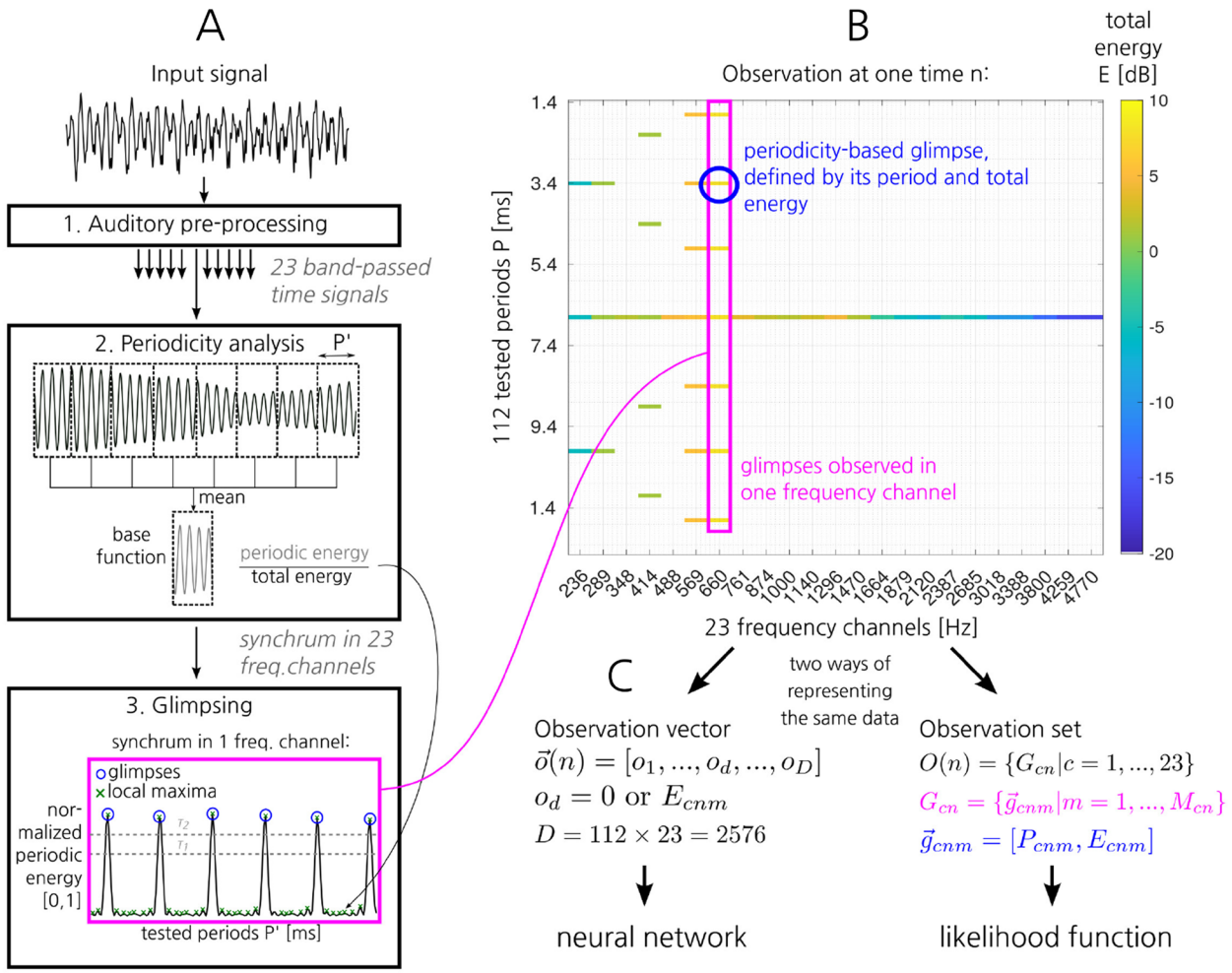


**Fig. 1** Extraction of sparse periodicity-based auditory features (sPAF). **A** Three main processing stages. **B** Glimpsed observation in one time instance. **C** Two ways of representing sPAF for different state estimation methods. Image based on [4]

Above, the sPAF were defined as a set $O(n)$ with a varying number of salient components. While such representation can be used as an input to an analytically designed model from Section 2.3, it is not suitable for training a neural network that requires a fixed dimensionality of the input features. To present the features to the neural network, sPAF can be treated as a 2-dimensional matrix of size $112 \times 23$, representing 112 tested period values in 23 frequency channels. Such a matrix can be reshaped into an observation vector with a dimensionality $D = 112 \times 23 = 2576$ . Entries of the matrix for which a glimpse was found are set to the glimpse energy value $E_{cnm}$. All remaining entries are set to 0.

### 2.2 State estimation methods

Within a time frame of 20-ms duration, speech is usually considered to be stationary. For every frame number $n$, a continuously voiced signal can be characterized by a *state vector* containing the values of the instantaneous fundamental frequency $F0$ and first two formant frequencies $F1$ and $F2$:

$$\vec{s}(n) = \begin{pmatrix} F0(n) \\ F1(n) \\ F2(n) \end{pmatrix}. \tag{5}$$

In this study, we considered voiced signals, for which continuous three-dimensional state trajectories can be defined (see Fig. 6).

The objective of the models presented in this study is to infer voice state parameters $\vec{s}(n)$ based on the *observation* $O(n)$ containing sparse periodicity-based auditory features (sPAF). Note that while the state is defined as a vector with a fixed dimensionality, the observation is a set

in which the number of elements depends on the acoustic signal (for more details, see Section 2.1). The sections below outline the state estimation methods compared in this study. For a theoretical introduction to state-space methods and Monte Carlo sampling, the reader is referred to [9, 10, 15].

#### 2.2.1 Non-sequential Monte Carlo sampling

In a Bayesian framework, the state estimation can be formulated as finding the *posterior* probability of a current state given the current observation. According to Bayes' rule, this probability can be computed as:

$$p(\vec{s}(n)|O(n)) = \frac{p(O(n)|\vec{s}(n))p(\vec{s}(n))}{\int p(O(n)|\vec{s}(n))p(\vec{s}(n))d\vec{s}(n)}, \tag{6}$$

where the *likelihood* $p(O(n)|\vec{s}(n))$ describes variation in the data for a fixed state and the *prior* $p(\vec{s}(n))$ describes prior beliefs about the possible state. Even if the likelihood and prior can be evaluated to give an unnormalized posterior, the integral in the denominator of the equation above is usually intractable. An analytic closed-form expression is, therefore, not available. However, it is possible to approximate posteriors or expectation values w.r.t. posteriors. We here use a Monte Carlo approach based on samples with normalized importance weights (e.g., [10]) in order to represent the posterior, and in order to compute corresponding expected values. Concretely, in each time step, we draw 2000 three-dimensional samples (hypothetical states) from a uniform prior distribution in 3 dimensions ($\mathcal{U}(100, 400)$ for $F0$, $\mathcal{U}(300, 800)$ for $F1$, and $\mathcal{U}(800, 2500)$ for $F2$). Next, the weight for each of the three-dimensional hypothetical states is computed by evaluating the $p(O(n)|\vec{s}(n))$, which is designed to capture the relationship between the voice parameters F0, F1, and F2 and the observed sPAF (see Section 2.3.2). $p(O(n)|\vec{s}(n))$ is a probabilistic function that takes sPAF data and hypothetical voice parameters as input arguments and outputs a likelihood weight value. Weights for 2000 state samples are normalized so that they sum to 1. The final estimate $\hat{\vec{s}}(n)$ is the three-dimensional state which maximizes the posterior.

#### 2.2.2 Sequential Monte Carlo sampling

The state estimation method introduced in this section was previously used in [4, 14] to simulate the process of attentive voice tracking. The sequential Bayesian state estimation is formulated as finding the posterior probability of a current state given the sequence of previous observations:

$$p(\vec{s}(n)|O(0:n)) = \frac{p(O(n)|\vec{s}(n))p(\vec{s}(n)|O(0:n-1))}{p(O(n)|O(0:n-1))}, \tag{7}$$

where

$$p(\vec{s}(n)|O(0:n-1)) = \int p(\vec{s}(n)|\vec{s}(n-1))p(\vec{s}(n-1)|O(0:n-1))d\vec{s}(n-1), \tag{8}$$

$$p(O(n)|O(0:n-1)) = \int p(O(n)|\vec{s}(n))p(\vec{s}(n)|O(0:n-1))d\vec{s}(n). \tag{9}$$

Similarly to the non-sequential case, analytical evaluation is usually not possible, and an approximation is needed. Sequential Monte Carlo sampling, also called *particle filtering*, is a broadly used sampling method, which is used for sequentially estimating the posterior density [9]. The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples and weights. The main difference is that the new samples depend on the previous samples and that the weight propagates across time steps. The relationship between two subsequent states is described by the *state transition model* $p(\vec{s}(n)|\vec{s}(n-1))$.

Specifically, the particle filter iteratively executes the following steps:

1. *Initialization*: At the system onset, 2000 hypothetical state samples are drawn from a prior distribution, which is a normal distribution centered around the true state value.
2. *Prediction*: Particle filter iteration begins with predicting new state samples given the previous state samples via the state transition model.
3. *Update*: Hypothetical states are evaluated using incoming observation. For each sample, the weight is computed by multiplying the likelihood $p(O(n)|\vec{s}(n))$ with the old weight.
4. *Estimation*: The final estimate $\hat{\vec{s}}(n)$ is computed as expected value from the approximated posterior, i.e., discrete distribution of state samples with corresponding weights.
5. *Resampling*: To direct the hypotheses set into the region of high importance, the samples with small weights are eliminated, and the samples with large weights are duplicated. After this step, the new iteration begins.

Likelihood models used in the above procedure are detailed in Section 2.3.

#### 2.2.3 Regression neural network

If we assume the states $\vec{s}(n)$ given an observation $\vec{o}(n)$ to be well approximated by a deterministic function $f(\cdot)$, i.e.,

$$\vec{s}(n) = f(\vec{o}(n)), \tag{10}$$

then the function $f(\cdot)$ can be approximated using data-driven approaches (if sufficient data is available). Here we use a regression neural network to approximate the function $f(\cdot)$. Given a finite number of input and output pairs $(\vec{o}(n), \vec{s}(n))$ for training, we take the output $f(\vec{o}(n))$ during test-time to approximate the most likely combination of state parameters $F0$, $F1$, $F2$ for the observed sPAF.

*Network and training* We used a regression neural network to learn the relationship between sPAF patterns $\vec{o}(n)$ and corresponding voice parameters $\vec{s}(n)$. We used a standard feed-forward regression neural network with an input layer of 2576 neurons, matched with the dimensionality of an observation vector. The network has two fully connected hidden layers with 1000 and 100 neurons respectively. The activations for all layers apart from the output are sigmoid functions. The output layer has a dimensionality matched with the state vector and a linear activation function. The network was trained using the *Nadam* optimizer [16] and *log-cosh* regression loss function, with 100 training epochs. The training data set consisted of pairs of data points (observation vectors $\vec{o}$) and corresponding labels (state vectors $\vec{s}$). Input feature vectors $\vec{o}(n)$ were normalized to values between 0 and 1. Target parameters ($F0$, $F1$ and $F2$) were scaled using the global mean and standard deviation, so that each parameter can only take values between 0 ad 1. After training, the inferred parameter values were scaled back to original value ranges.

To generate the training pairs, we created random 3-dimensional state trajectories with a sampling rate of 50 Hz. They were used as an input to the Klatt formant synthesizer [17], yielding synthetic voice signals. The instantaneous sPAF were extracted from the acoustic signal with the same sampling rate as the trajectory.

Two different training data sets were used in the study:

1. Clean sPAF: data set generated based on sPAF extracted from 1000 voice signals of 10 s each, in total 501,000 state-observation pairs.
2. Clean and fragmentary sPAF: data set generated based on clean voice sPAF, artificially removed sPAF, and segregated sPAF (for details, see Section 3). In each category, 1000 trajectories of 2 s each, in total 404,000 state-observation pairs.

### 2.3 F0, F1, F2-models

This section reviews probabilistic models required for the Bayesian Monte Carlo state estimation methods (Sections 2.2.2 and 2.2.1), specifically for estimating instantaneous voice parameters ($F0$, $F1$, and $F2$) based on sPAF.

#### 2.3.1 F0, F1, F2-transition model $p(\vec{s}(n)|\vec{s}(n-1))$

The state transition model describes the temporal evolution of parameters $F0$, $F1$, and $F2$, which are naturally limited due to physical constraints of speech production. For simplicity, we assume independence of the individual dimensions; therefore, subsequent values in each dimension are predicted individually.

To predict the next value for the $i$-th state dimension $s_i(n)$, the trend $\Delta\widehat{s_i}(n) = \widehat{s_i}(n-2) - \widehat{s_i}(n-1)$ between two previous estimates is calculated, the next value according to that trend $s_i(n) + \Delta\widehat{s_i}(n)$ is predicted, and finally, gaussian noise is added to the predicted value $s_i(n) + \Delta\widehat{s_i}(n) + \epsilon_i$, where $\epsilon_i = \mathcal{N}(\mu = s_i(n), \sigma = \sigma_i)$, and $\sigma_i$ is 0.5 Hz for $F0$, 1 Hz for $F1$, and 5 Hz for $F2$. In addition, we make sure that the difference between two previous estimates $\Delta\widehat{s_i}(n)$ does not exceed the largest allowed step (10 Hz for F0, 50 Hz for F1, and 100 Hz for F2) and that the extrapolated value $s_i(n) + \Delta\widehat{s_i}(n)$ does not exceed a possible value range ([100, 400] for F0, [300, 800] for F1, and [800, 2500] for F2). Figure 2 depicts the procedure, repeated for every state sample.

#### 2.3.2 F0, F1, F2-observation model $p(\boldsymbol{O}(\boldsymbol{n})|\vec{\boldsymbol{s}}(\boldsymbol{n}))$

The $F0, F1, F2$—observation model is a probabilistic function that relates the observed sPAF and the underlying voice parameters. It quantifies the likelihood that the sPAF $O(n)$ extracted in a given time frame come from a hypothetical three-dimensional voice state $\vec{s}(n)$. There are two major assumptions, which influence the design of this function. First, we assume that the glimpses in one channel $G_{cn}$ originate from a single voice, even if the acoustic signal contains a mixture of voices. This *saliency assumption* is based on the fact that we use the glimpsing thresholds, which ensure that the glimpses are extracted only if one voice dominates in the signal. This is demonstrated in Fig. 3.

Secondly, we assume that the state dimensions are independent and that period $P_{cnm}$ is solely the evidence of $F0$ and energy $E_{cnm}$ is solely the evidence of $F1$ and $F2$.

Hence, the likelihood of a single glimpse $p\left(\vec{g}_{cnm}|\vec{s}(n)\right)$ can be approximated as follows:

$$p\left(\vec{g}_{cnm}|\vec{s}(n)\right) \approx p(P_{cnm}|F0(n)) \cdot p(E_{cnm}|F1(n), F2(n)), \tag{11}$$

where the approximation is motivated by the above stated assumption.

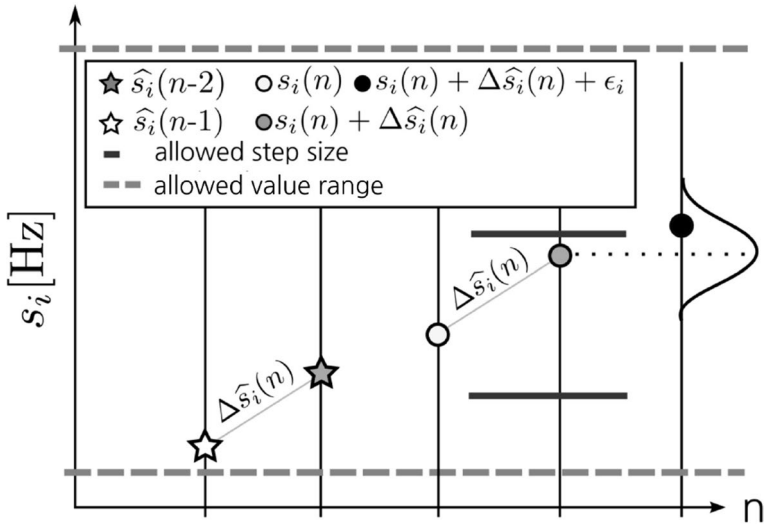


**Fig. 2** State transition probability model predicting the next state value. $s_i(n)$: hypothetical value of the state dimension i, $\widehat{s_i}(n-1)$: estimate in the last time step, $\widehat{s_i}(n-2)$: estimate in the second-to-last time step, $\Delta\widehat{s_i}(n)$: difference between the last two estimates. Allowed step size was 10 Hz for F0, 50 Hz for F1, and 100 Hz for F2. Possible value range was [100, 400] for F0, [300, 800] for F1, and [800, 2500] for F2. Image based on [4]

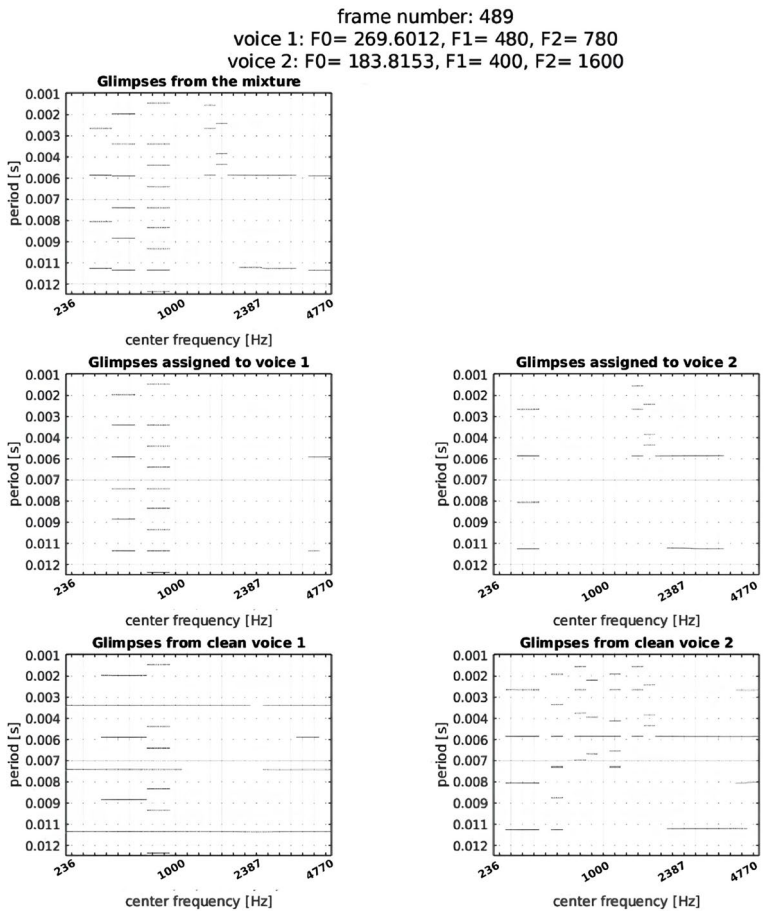


**Fig. 3** Example of sPAF in one time frame, which meet the saliency assumption. The top panel shows sPAF extracted from a mixture of two voices, the middle panels show the same sPAF segregated into two voices (each channel was assigned to either of the voices). The lower panels show sPAF extracted from individual voices before mixing. Each non-empty channel of the top panel represents either of the two voices

See Fig. 4 for a scheme demonstrating the procedure to evaluate this function. We call $p(P_{cnm}|F0(n))$ the *period likelihood* and $p(E_{cnm}|F1(n), F2(n))$ the *energy likelihood*.

The likelihood that an observed glimpse energy $E_{cnm}$ originates from a hypothetical $F1$ and $F2$ is modeled using two normal distributions with mean and standard deviation defined by the two codebooks for $F1$ and $F2$ as follows:

$$p(E_{cnm}|F1(n), F2(n)) = \mathcal{N}\Big(E_{cnm}; \mu_E(F1, c), \sigma_E^2(F1, c)\Big) \cdot \mathcal{N}\Big(E_{cnm}; \mu_E(F2, c), \sigma_E^2(F2, c)\Big). \tag{12}$$

*Energy likelihood* $p(E_{cnm}|F1(n), F2(n))$ For the energy likelihood, we used a codebook approach to evaluate this function. The codebook entries were computed from the simulated data: First, a list of $F1$ values logarithmically spaced between 350 and 700 Hz, and a list of $F2$ values logarithmically spaced between 800 and 2500 Hz was created. Next, for each $F1$ value, a 10-s signal with fixed $F1$ and varying $F0$ and $F2$ was generated. Likewise, for each $F2$ value, 10-s signal with fixed $F2$ and varying $F0$ and $F1$. sPAF were extracted from the signal and for every $F1$ a mean glimpse energy $\mu_E(F1, c)$ and a standard deviation $\sigma_E(F1, c)$ was computed and stored. Similarly for every $F2$ a mean energy $\mu_E(F2, c)$ and a standard deviation $\sigma_E(F2, c)$ was computed and stored.

We also experimented with other models for the energy likelihood including non-factorizing likelihoods or differently parameterized codebooks. The above modeling was finally chosen based on its relative performance, stability, and efficiency benefits. It should be noted that the codebook summarizes energy distribution for only one input signal level. In order to account for voice signal level fluctuations, the procedure would need to be repeated for multiple input levels.

*Period likelihood* $p(P_{cnm}|F0(n))$ The likelihood that an observed glimpse period value $P_{cnm}$ originates from a given $F0$ is modeled using a mixture of circular von-Mises distributions [see 4]. Every value $P_{cnm}$ is generated

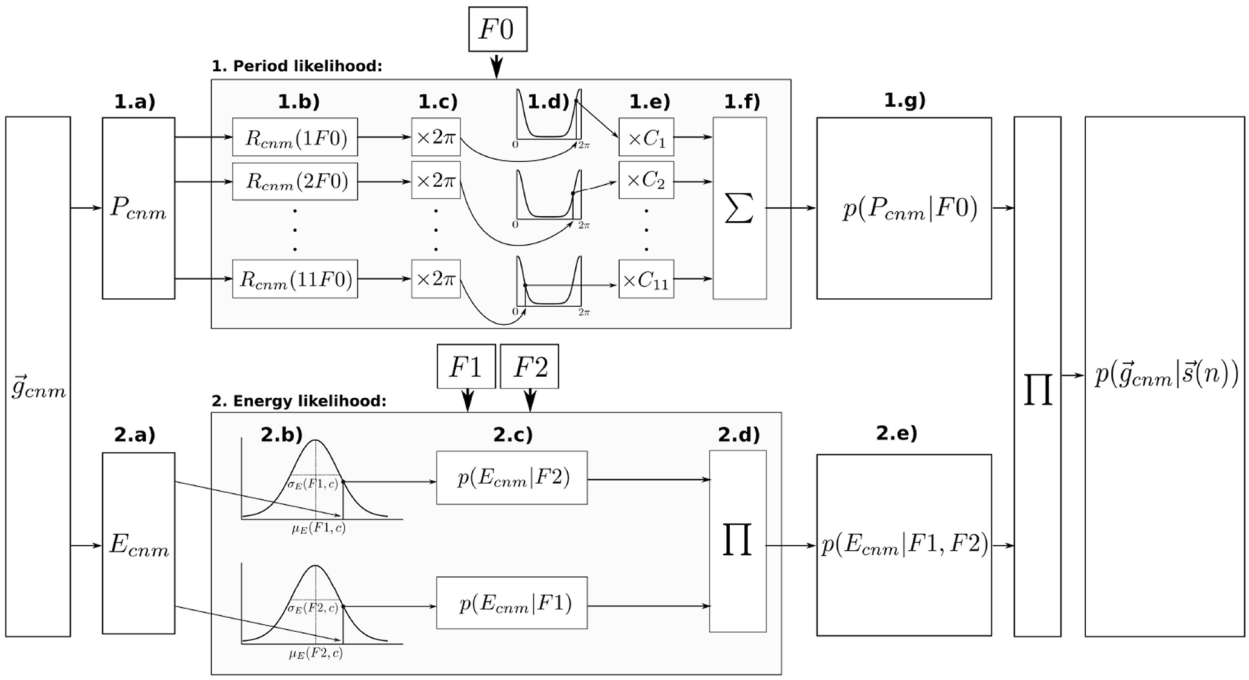


**Fig. 4** Evaluating glimpse likelihood. (1) Procedure for evaluating period likelihood given $F0$. Based on a single observed period value (1.a), 11 relative period values $R_{cnm}(jF0)$ are computed, where $j$ is the harmonic number (1.b). Next, they are multiplied by $2\pi$ to obtain a circular variable (1.c). The resulting 11 values are evaluated with the circular von-Mises distribution centered at 0 (1.d). Each likelihood is multiplied with a normalizing constant, which depends on the harmonic number (1.e). The values are added (1.f) and the final result is the likelihood of a period value given hypothetical $F0$ (1.g). (2). Procedure for evaluating energy likelihood given $F1$, $F2$. To obtain energy likelihood given $F1$, the observed energy value (2.a) is evaluated with a normal distribution (2.b) whose parameters are taken from a codebook storing mean and standard deviation for different $F1$ values. The same is repeated for $F2$ to obtain the energy likelihood given $F2$. The energy likelihoods for $F1$ and $F2$ (2.c) are multiplied (2.d) to obtain joint likelihood (2.e). The final glimpse likelihood given a hypothetical state ($F0$, $F1$, $F2$) is the product of period and energy likelihoods

by a mixture of 11 circular von-Mises distributions. The number 11 comes from the highest reported number of resolved harmonics [18]. Each element of the sum represents a different harmonic $j$ of $F0$:

$$p(P_{cnm}|F0) = \sum_{j}^{11} C_j \mathcal{M}\big(R_{cnm}(j \cdot F0) \cdot 2\pi; \mu, \kappa\big), \tag{13}$$

where $F0$ is the hypothetical fundamental frequency, $\mathcal{M}$ denotes von-Mises distribution with the mean $\mu = 0$ and concentration parameter $\kappa = 5$. $C_j = \frac{j^{-1}}{\sum_{j'}^{11} j'^{-1}}$ is the normalizing constant for the $j$-th harmonic. It is reciprocal to harmonic number: the higher the harmonic number, the lower the probability of the period glimpse originating from that harmonic. $R_{cnm}(j \cdot F0)$ is the relative period value with respect to the $j$-th harmonic of the hypothetical $F0$ and is computed as:

$$R_{cnm}(F0) = \mathrm{rem}(\frac{\mathrm{P_{cnm}}}{\mathrm{P0}}) = \mathrm{rem}(\mathrm{P_{cnm}} \cdot \mathrm{F0}), \tag{14}$$

where $P0 = F0^{-1}$ is the period of $F0$ and $\mathrm{rem}(\cdot)$ is the remainder from the division.

For a detailed explanation of the period likelihood function, the reader is referred to our recent study [4].

*Likelihood integration* In each non-empty channel set $G_{cn}$ , the likelihood is integrated by computing a mean across the likelihoods of the elements of the channel set:

$$p(G_{cn}|\vec{s}(n)) = \frac{1}{M_{cn}} \sum_{m} p\big(\vec{g}_{cnm}|\vec{s}(n)\big). \tag{15}$$

Finally, the likelihood is integrated as a product across frequency channels:

$$p(O(n)|\vec{s}(n)) = \prod_{c} p\Big(\vec{G}_{cn}|\vec{s}(n)\Big). \tag{16}$$

## 3 Evaluation

Four different state estimation methods were compared:

1. Non-sequential Monte Carlo sampling, denoted *nonseqMC*
2. Sequential Monte Carlo sampling denoted *seqMC*
3. Regression neural network trained with clean voice sPAF (Section 2.2.3), denoted *regNN*
4. Regression neural network trained with fragmentary sPAF (Section 2.2.3), denoted *regNN+*

Figure 5 illustrates the above listed methods. The performance of each method was evaluated under the following conditions:

a) *Clean voice sPAF*: features were extracted from the acoustic signal containing one voice.
b) 40% *Artificially removed sPAF*: 40% of glimpses were artificially removed from the clean voice sPAF, with uniform removal probability for all channels and time steps.
c) 80% *Artificially removed sPAF*: 80% of glimpses were artificially removed from the clean voice sPAF, with uniform removal probability for all channels and time steps.
d) *Optimally segregated*: Features were extracted from the acoustic signal containing two voices and segregated using an F0-based feature segregation method (see the Appendix "Optimal feature segregation method" section for details). sPAF assigned to the considered voice were used in the state estimation.

Figure 6 depicts the above listed test conditions. Test data were obtained using the same procedure as in [4, 5]: First, 100 random 3-dimensional state trajectories each of length $L = 100$ were created. The trajectory of each parameter ($F0$, $F1$, $F2$) was generated independently, by picking a random excerpt of Gaussian noise (500 Hz sampling rate), filtering it between 0.05 and 0.6 Hz and adjusting the value range to $100 - 400$ Hz for $F0$, $300 - 800$ Hz for $F1$, and $700 - 2200$ Hz for $F2$. The trajectories with a sampling rate of 50 Hz were used as an input to the Klatt formant synthesizer [17], yielding 2 s-long synthetic voice signals, from which the sPAF were extracted with the same sampling rate. Conditions with artificially removed glimpses approximate adding noise to the signal before extracting sPAF. For example, 40% and 80% of glimpse loss corresponds to adding white noise to a voiced signal at $15 - 20$ dB SNR, and $0 - 5$ dB SNR, respectively. In the condition using a mixture of 2 voices (optimally segregated), a second set of 100 acoustic signals was synthesized and mixed with the single voice signals before feature extraction. In the case of state estimation with a regression neural network, the sPAF features were additionally transformed to a suitable format with a fixed dimensionality (for details, see Section 2.1). For test conditions with fragmentary data (b–d), this resulted in more zero-entries in the observation vector.

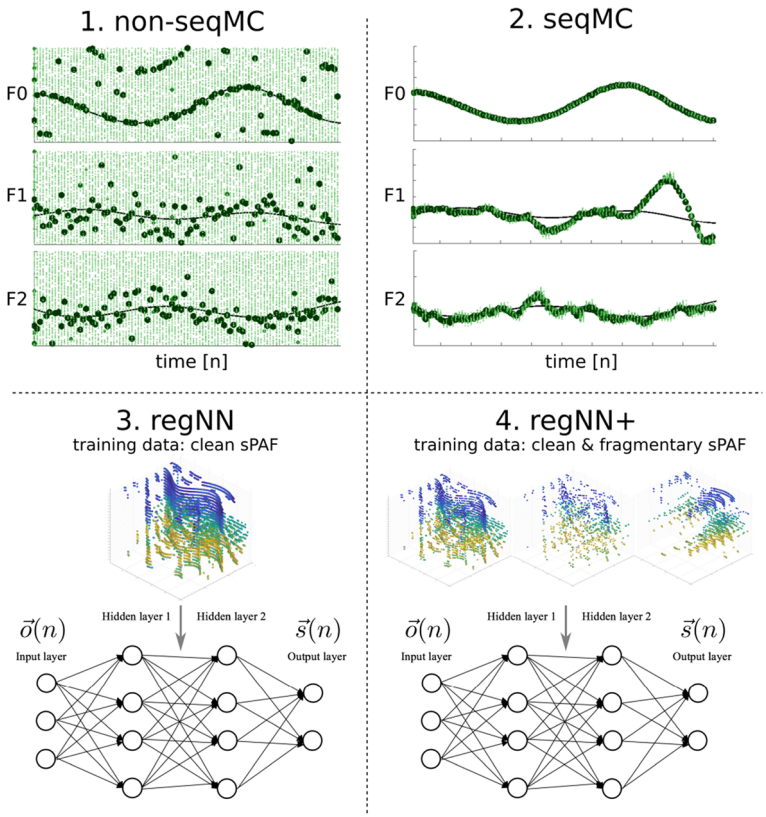


**Fig. 5** State estimation methods. **1** Non-sequential Monte Carlo simulation: 3 plots show state estimation procedure in each of the 3 dimensions: *F*0, *F*1, *F*2. Dots represent hypothetical states. Color intensity and the size represent the likelihood $p(O(n)|\vec{s}(n))$ computed for each state sample, given the input sPAF data $O(n)$. The state is estimated independently in each time step. **2** Sequential Monte Carlo simulation: 3 plots show state estimation procedure in each of the 3 dimensions: *F*0, *F*1, *F*2. Dots represent hypothetical states (particles). Color intensity and the size represent the weight computed via likelihood $p(O(n)|\vec{s}(n))$ for each state sample, given the input sPAF data $O(n)$. Continuity model $p(\vec{s}(n)|\vec{s}(n-1))$ defines how the samples evolve between consecutive time steps. Due to resampling, particles are focused on the region of the highest importance. **3** Regression neural network trained with the clean sPAF: trained network generates the most likely output state $\vec{s}(n)$, given the observation $\vec{o}(n)$. **4** Regression neural network trained with the fragmentary sPAF (40% removed, 80% removed, and optimally segregated): trained network generates the most likely output state $\vec{s}(n)$, given the observation $\vec{o}(n)$. $\vec{o}(n)$ is the transformed version of the $O(n)$

The following performance measures were used to compare the state estimation with different methods:

1 *Root mean square error (RMSE)*: $\sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\vec{s}_i(n)-\hat{\vec{s}}_i(n)\right)^2}$, where $N$ is the cumulative length of all trajectories.
2 *Gross error*: percentage of time steps for which the estimate lies outside the allowed interval (5 Hz for *F*0, 25 Hz for *F*1, and 100 Hz for *F*2).
3 *Harmonic error*: model-based similarity measure between tracks of fundamental frequency (for details, see the Appendix "Harmonic error" section).

## 4 Results

This section presents the results for four state estimation methods in five conditions with different types of input features.

Figure 7 shows examples of ground truth *F*0, *F*1, and *F*2 trajectories together with the estimated trajectories for all methods and conditions.

The results are analyzed and discussed for each feature dimension. Figure 8 shows performance measures computed for the first dimension of the state space: *F*0.

Non-seqMC method results in the highest *F*0 RMSE of all methods, and in all conditions. The values are similar across different feature conditions and reach 92.8 Hz, which exceeds the RMSE computed for the white noise with mean 250 Hz and standard deviation 50 Hz. This indicates a systematic error leading to very low performance in terms of absolute estimation accuracy. Examples in Fig. 7 (1.a–1.d) demonstrate that, while the estimated values are indeed far off from the underlying values, the errors are not random and are caused by the

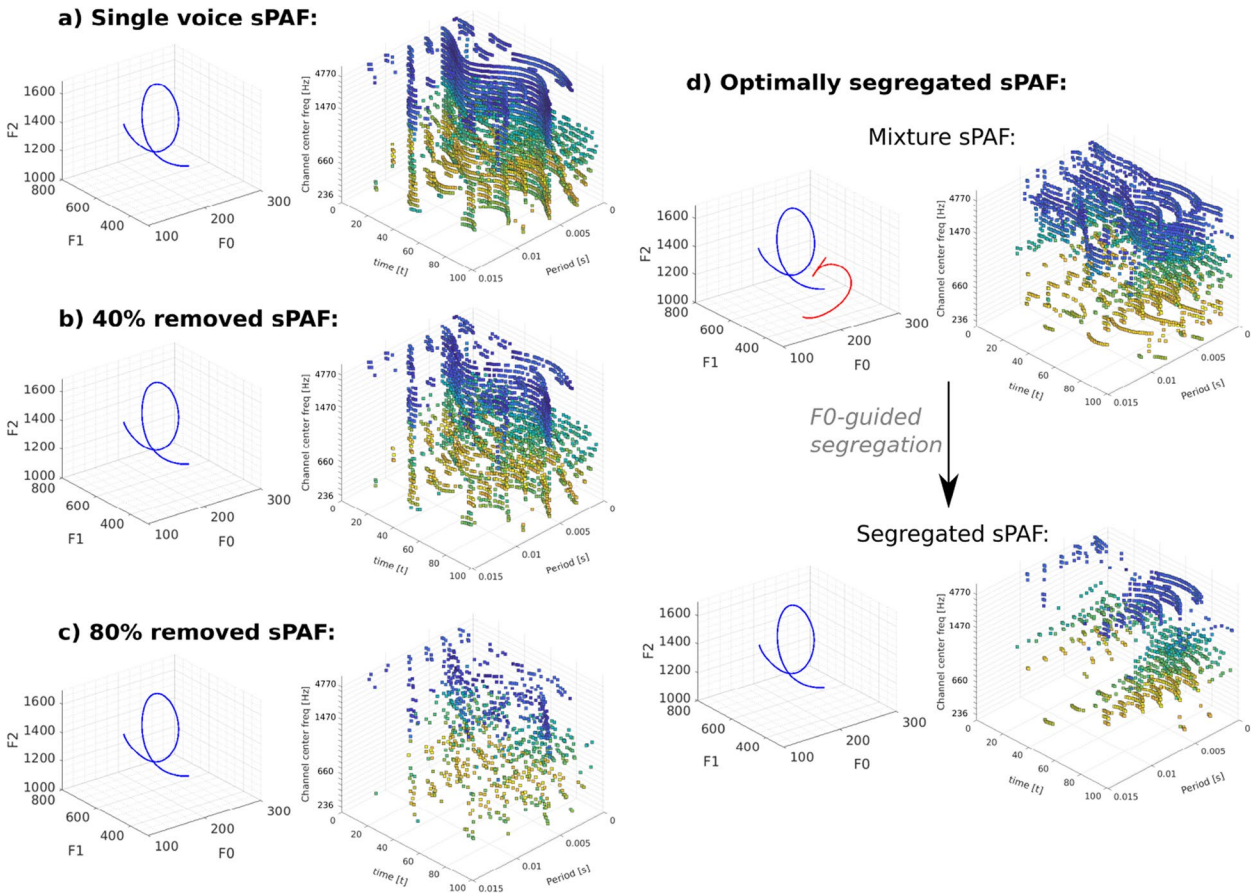


**Fig. 6** Testing conditions. sPAF are depicted together with the hidden state trajectories. **a** sPAF extracted from a clean synthetic voiced signal. **b** 40% of non-empty glimpse channels removed from the sPAF extracted from a clean synthetic voiced signal. **c** 80% of non-empty glimpse channels removed from the sPAF extracted from a clean synthetic voiced signal. **d** 1 voice features segregated from the sPAF extracted from a mixture of two synthetic voiced signals

*F*0 harmonic confusions typical for pitch tracking. This observation is confirmed by harmonic error, which is close to the seqMC and is lower than the harmonic error for regNN+. This suggests that the errors of the non-seq MC are mainly caused by harmonic confusion. Similar observation can be made for the seqMC method in the segregated sPAF condition—the increased RMSE is caused mostly due to harmonic confusions, which can be seen in the Example in Fig. 7 (2.d).

As seqMC can avoid harmonic confusion, it reaches the best performance of all methods, in most conditions (see examples in Fig. 7 (2.a and 2.d)). This confirms that limiting the possible outcomes by adding the expectation component in the state estimation is crucial for its performance. The only exception is the clean sPAF condition, where the regDNN, trained with the clean data, achieves the lowest errors.

A much different relationship between the error values and feature types can be observed for the regDNN method. The lowest errors are observed for the clean sPAF, with RMSE of 6.6 Hz. Performance decreases to 16.1 Hz after removing 40% of sPAF and to 37.6 Hz after removing 80%. The difficulty in the feature conditions has a significant effect on the regNN performance. Although for clean sPAF the model outperforms all other methods, the benefit from regNN decreases for all types of sPAF features, which were not included in the training. This shows that this model is not capable of generalizing well for fragmentary information.

The regDNN+ method achieves similar results across all feature conditions. Performance decreases slowly from 10.7 Hz in the clean sPAF condition to 39.9 Hz in the segregated condition. This shows that the network trained with various types of fragmentary information achieves on average good results for all feature types, at the cost of precise results in the clean sPAF.

Figure 9 shows performance measures computed for the second dimension of the state space: *F*1.

The best overall *F*1 estimation performance is achieved by the regNN+ method—a method that leads to only

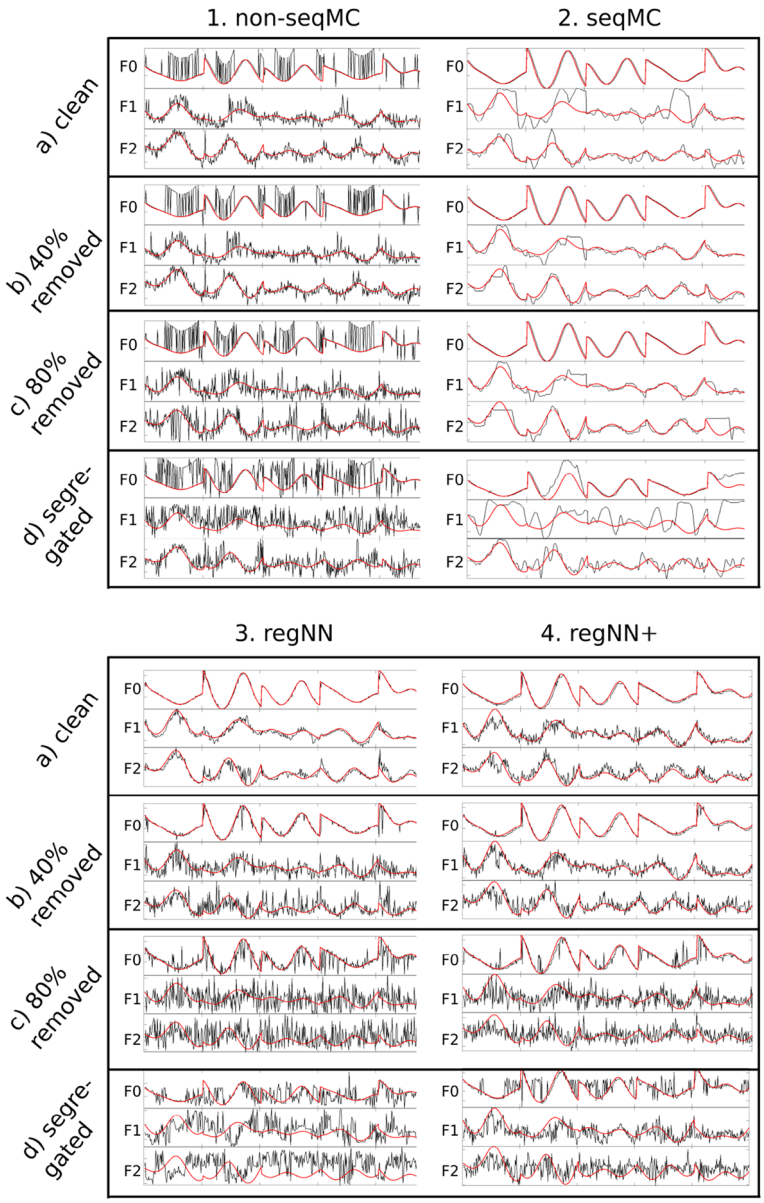


**Fig. 7** Example of a chosen excerpt of ground truth parameter trajectories (red) plotted together with the estimated parameter trajectories (black) for all methods and feature conditions

average performance for the $F0$ estimation. RegNN+ obtains the RMSE of $37.8 - 76.7$ Hz and outperforms seqMC in most conditions. It can approximate the relationship between sPAF and $F1$ more precisely than the seqMC method. The energy likelihood model used in the Monte Carlo methods assumes that the glimpsed energy is only the evidence of $F1$ and $F2$ and that the period values are only the evidence of $F0$. This assumption might not be valid, especially in the low frequency channels, where most evidence for $F1$ can be found: on the one hand, the energy in those channels is influenced by $F0$, and on the other hand, the observed period harmonics depend on the spectral filtering dictated by the formants. This simplistic energy likelihood model is most likely valid only for the higher frequency channels that provide less evidence for $F1$. This can explain the poor $F1$ estimation performance of the Monte Carlo methods. regNN+ trained with the fragmentary information is not bound by such assumptions and can capture a more complex relationship between the sPAF patterns and $F1$.

A similar problem with the energy likelihood model is manifested as a large error difference between the 80%

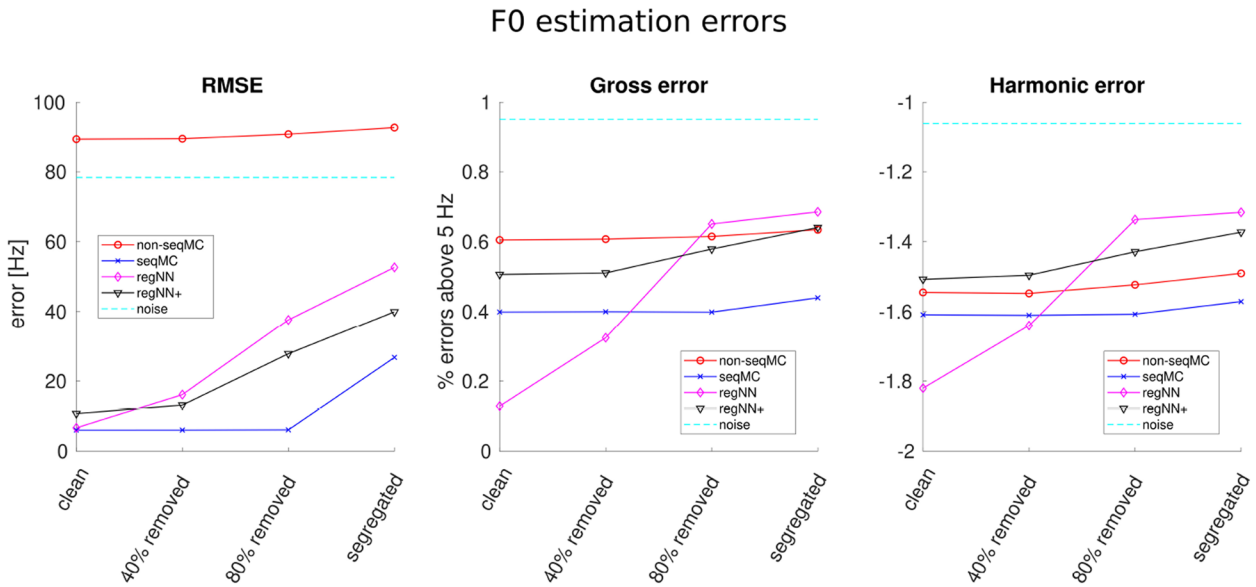


**Fig. 8** Performance measures for *F*0 estimation. *y*-axis: error measure, *x*-axis: different input features, solid lines: different state estimation methods, dashed line: error computed for the artificially generated white noise with mean 250 Hz, and standard deviation 50 Hz

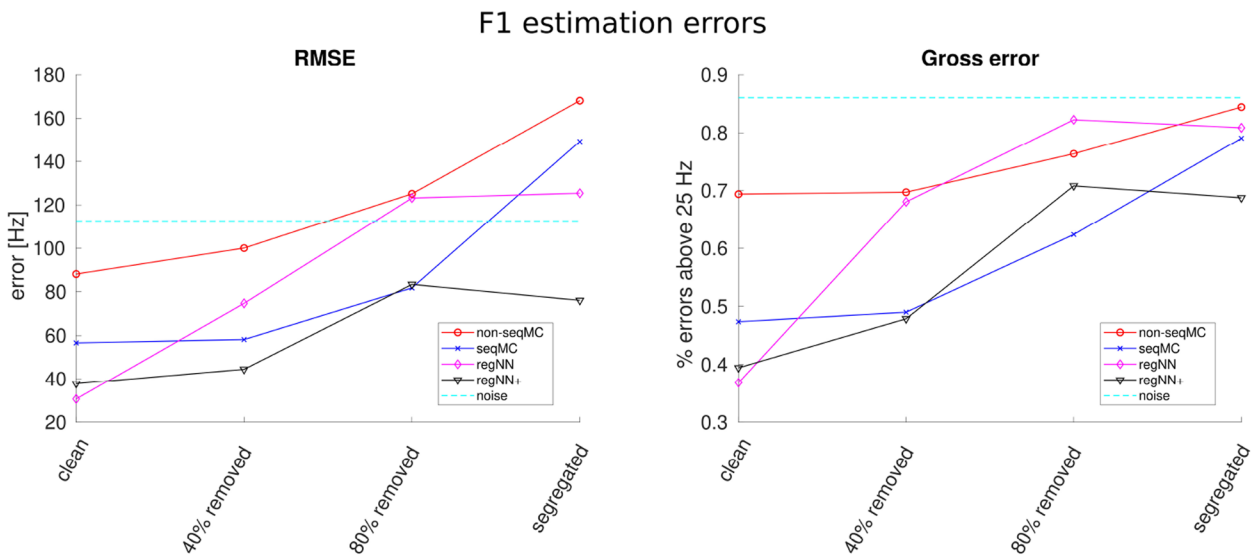


**Fig. 9** Performance measures for *F*1 estimation. *y*-axis: error measure, *x*-axis: different input features, solid lines: different state estimation methods, dashed line: error computed for the artificially generated white noise with mean 550 Hz, and standard deviation 83 Hz

removed sPAF and the segregated sPAF. In the first condition, the glimpses are removed from clean sPAF with equal probability for all channels. In the second condition, glimpses are removed depending on the second voice in the mixture. If the segregation removes the only frequency channels that can be interpreted by the energy likelihood model, the model will fail to estimate *F*1.

For the clean sPAF, the best performance is again achieved by the regNN, which was trained for the clean data. Here again, the performance drops drastically after removing some information from the sPAF, indicating that the model is overfitted to the clean sPAF.

All methods besides regNN+ for some conditions exceed the RMSE level computed for the noise (normally

distributed with mean 550 Hz, and standard deviation 83 Hz). Above this level, we can consider the method unable to estimate $F1$. The failure of the Monte Carlo methods is most likely caused by the over-simplified energy model, and the failure of the regDNN is caused by the inability to generalize for the unseen data.

Figure 10 shows performance measures computed for the third dimension of the state space: $F2$.

As in the previous two dimensions, regNN is overfitted to the clean sPAF. It achieves the best results of all for clean sPAF (RMSE of 99.98 Hz), but the worst results of all in all other conditions. The best performance is achieved by the seqMC method; however, unlike in the $F0$ estimation, the errors increase with the difficulty in the feature condition. SeqMC is significantly better than the remaining methods (non-seqMC and regDNN+), which do not use expectation in the estimation process.

## 5 Discussion

In this study, we compared the voice state estimation performance of two classes of state estimation methods: Bayesian sampling and deep learning. The first class uses analytically formulated probability models. They evaluate the data likelihood for a finite hypotheses set, thus approximating state posterior distribution, based on which the most likely state can be estimated. The second class approximates the mapping between the hidden state and the data in a supervised learning procedure. A trained model allows for predicting the most likely state for a given data vector. As presented in this work, both approaches can be used for voice parameter estimation. However, there are interesting differences in the performance of these methods for different types of input sPAF.

To understand these differences, it is useful to quickly review the main objectives of the approaches. Probability models used in Monte Carlo simulations are designed to describe specific properties of the sound. They provide a concise explanation of the relationship between the observed data and the hidden parameters. They are interpretable but use assumptions that limit their complexity. In contrast to that, the objective of deep learning is to precisely approximate this relationship. They provide limited interpretability but can model complex non-linear dependencies. This demonstrates the different nature of these approaches. The question we posed in this study is how these two (to a certain extend contradictory) powers can be used to interpret fragmentary data.

$F0$ estimation performance with Monte Carlo methods, which used period likelihood formulation from [4], is least influenced by the changes in sPAF, which proves that the observation model can generalize well across several conditions with fragmentary information. It suggests that the period likelihood model accurately describes the properties of a single voice, and it might be better suited to model human performance.

In contrast to $F0$ estimation, formant estimation does not benefit so much from the analytical modeling approach. Especially for $F1$, the simplistic model of the energy likelihood does not seem to sufficiently capture the relationship between the observed energy glimpses and the parameters. $F1$ and $F2$ estimation performance

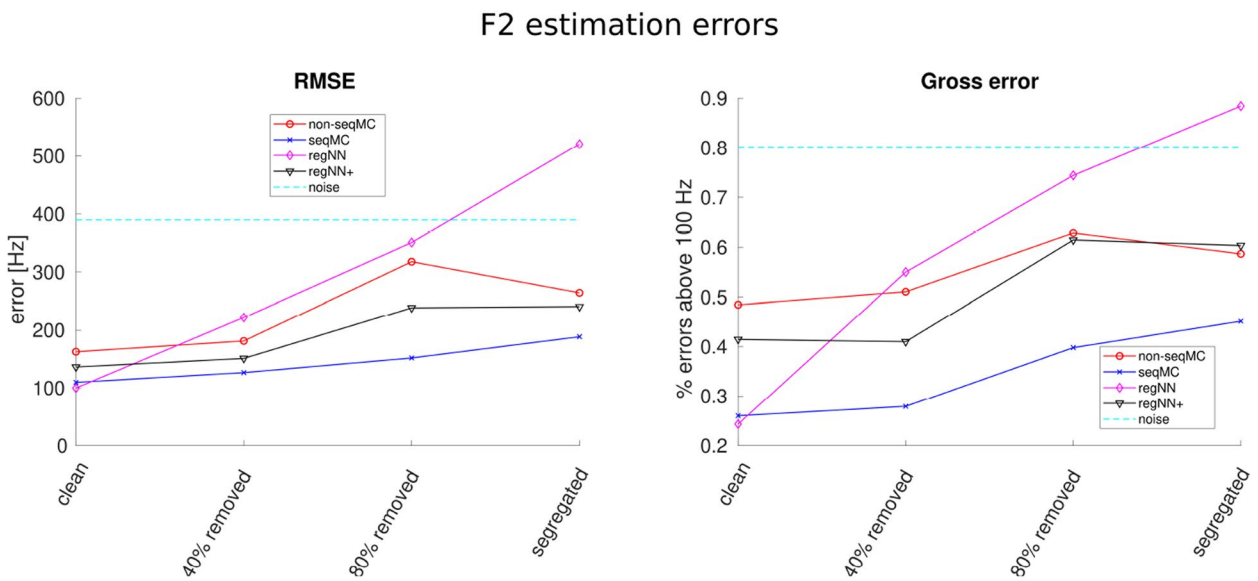


**Fig. 10** Performance measures for $F2$ estimation. $y$-axis: error measure, $x$-axis: different input features, solid lines: different state estimation methods, dashed line: error computed for the artificially generated white noise with mean 1600 Hz, and standard deviation 300 Hz

of Monte Carlo methods is sensitive to the feature types, meaning that the model is not capable of inferring the formants based on fragmentary information without losing accuracy.

While Monte Carlo methods depend on the quality of the probabilistic model, the neural networks are highly dependent on the training data. As expected, the network trained using clean sPAF alone leads to a model overfitted to this condition, i.e., the best results are observed in the clean condition but it is not able to generalize well to other conditions. The network trained using fragmentary sPAF is much less sensitive to the changes in sPAF. It leads to only average performance in $F0$ and $F2$ dimensions but is particularly beneficial for estimating $F1$: the network can learn dependencies between the state parameters, which are oversimplified in the probabilistic model used in MC methods.

The Sequential Monte Carlo method outperforms other methods in the ability to generalize across conditions with fragmentary information. The reason why the method is observed particularly capable is that the range of possible state outcomes is limited by the finite number of hypotheses that are updated in every time step. This property in combination with a valid observation model is what makes the performance independent of the degree of sparseness in the data. From the perspective of computational auditory scene analysis, this result confirms that top-down processing is essential in complex auditory scenes. Because of sound superposition, only a limited amount of robust information about a specific sound object is available in such conditions. Inference based on this fragmentary information is ambiguous and some form of a top-down expectation is required to resolve this ambiguity. Our results show that the benefit from the top-down approach grows with the difficulty in the input features, which indicates that the top-down-processing is increasingly important for incomplete input features. A similar conclusion was made by [19] in a machine learning study that investigated network architectures with attention mechanisms and showed that the more random modifications in the input data, the more the model relied on the top-level information.

In [4], we argued that a particle filter with a resampling step, which allows for focusing the distribution of the top-down expectation on the regions of high importance, is a plausible model of attentive tracking of one of two competing voices. In this study, we took a closer look at the features of a single voice separated from a mixture. The results of both studies lead us to the conclusion that to effectively model selective attention in the auditory system, we need optimal feature segregation followed by a model which confronts the incomplete input features with the top-down expectation to infer the current state of the auditory object. Results presented here indicate that simple feed-forward networks are not well suited for this task in comparison to Monte Carlo (MC) methods coupled with analytical probability models.

Before favoring the MC approach, it should be highlighted that the simple non-recursive network architecture, which we chose for our experiments, represents only a small fraction of the ways deep networks could be used in the context of attention modeling. On the one hand, our experiments do not show how deep learning could be harnessed to solve even smaller sub-tasks of the auditory scene. For example, a neural network could replace the likelihood models in the Monte Carlo framework. One could imagine a model that predicts the likelihood of cooccurrence for the input pair of state and observation [20]. On the other hand, the current work did not consider substituting the whole Monte Carlo framework with more complex recurrent architectures allowing for modeling the sequential dependencies in the data [21] and top-down processing [19, 22–24].

Possible future research is likely to profit from combining the positive aspects of deep networks and probabilistic methods. Developments in this direction are the integration of deep networks and probabilistic models as, e.g., represented by variational autoencoders [25, 26]. Standard VAEs are, however, not modeling time dependence which is, of course, crucial for data as considered in this study (as here evident, e.g., by the differences between sequential and non-sequential Monte Carlo). Recent developments have, therefore, extended VAE approaches by including, e.g., Gaussian processes as priors for VAEs [27–30]. The research direction is new, and applications to acoustic data including complex acoustic scenes still have to be investigated. But the positive aspects observed for deep neural networks and probabilistic approaches as studied here could in principle be combined based on such novel developments.

For applications, any approach also has to consider efficiency alongside other performance measures, however. Monte Carlo approaches such as particle filtering are known for their considerable computational costs related to testing a large number of hypotheses, and also VAE approach rely on sampling which becomes more challenging if complex priors such as Gaussian processes are used. An appropriate balance between performance and efficiency is, therefore, likely to determine which setup is finally the most appropriate for complex acoustic scenes.

Our work builds upon the previously developed auditory model of attentive tracking. Some of the choices in the model's components were driven by the intention to mimic the auditory system, without immediate consideration for applied signal processing. Furthermore, in this study, we prioritized assessing the model's behavior

within highly controlled scenarios over evaluating its performance in more realistic conditions. As a consequence, from the results presented here, it is difficult to conclude how close the proposed model is to tracking voices in realistic scenes with more elements of natural speech, ambient noise or reverberation. Nevertheless, it is important to note that removing information from the input features can be just as challenging to the model as adding noise. While some audio processing systems have already been tested with the fragmentary speech information in the past [1, 2, 6, 31], this is, to our knowledge, the first attempt to present such data in a recursive voice tracking paradigm.

In our study, we have shown that even relatively simple analytical likelihood models describing sound properties, when coupled with top-down expectation, can deal with fragmentary observation better than standard feed-forward neural network. In general, we believe this highlights the importance of incorporating top-down processing in models of selective listening.

## Appendix

### Harmonic error

Harmonic error is an auditory-inspired *F*0 estimation performance measure. The task of this measure is to evaluate the distance between two compared *F*0 trajectories in a perceptually relevant way.

Various studies demonstrate ambiguity of pitch perception [32–35]. Any tonal sounds other than a pure tone, especially complex tones lacking some harmonics, are more or less ambiguous in pitch. Computational models of pitch and *F*0 tracking algorithms also reflect this property of sound and suffer from ambiguous *F*0 estimates [33, 36, 37]. Ambiguity does not mean randomness: the pitches evoked by a stimulus are in systematic relationships to each other (they lie at the harmonics and their submultiples). Based on this, we can conclude that some of the *F*0 estimation errors are caused by the inherent nature of sound; hence, the performance measure should penalize those types of errors less than errors due to lack of precision in the algorithm.

The *harmonic error* computes the error between the ground truth $F0_{GT}$ and estimated $F0_{EST}$ using likelihood ratios computed with the *F*0 observation model from [4]. Specifically, the following procedure is used:

1 For both compared *F*0 generate a set $\tilde{O}$ of 100 hypothetical period glimpses by sampling from the mixture of circular von-Mises distributions (see Sec. 2.3.2):

$$F0_{GT} \rightarrow \tilde{O}_{GT} = \left[P_{GT}^{(1)}, P_{GT}^{(1)}, ...P_{GT}^{(100)}\right]$$

$$F0_{EST} \rightarrow \tilde{O}_{EST} = \left[P_{EST}^{(1)}, P_{EST}^{(1)}, ...P_{EST}^{(100)}\right]$$

2 Compute likelihoods:

$$p(\tilde{O}_{GT}|F0_{GT}) = \frac{1}{100}\sum_{i=1}^{100} p\left(P_{GT}^{(i)}|F0_{GT}\right)$$

$$p(\tilde{O}_{GT}|F0_{EST}) = \frac{1}{100}\sum_{i=1}^{100} p\left(P_{GT}^{(i)}|F0_{EST}\right)$$

$$p(\tilde{O}_{EST}|F0_{EST}) = \frac{1}{100}\sum_{i=1}^{100} p\left(P_{GT}^{(i)}|F0_{GT}\right)$$

$$p(\tilde{O}_{EST}|F0_{GT}) = \frac{1}{100}\sum_{i=1}^{100} p\left(P_{GT}^{(i)}|F0_{GT}\right)$$

3 Compute harmonic error as:

$$E_{harm} = \frac{1}{2}\left(\frac{p(\tilde{O}_{GT}|F0_{EST})}{p(\tilde{O}_{GT}|F0_{GT})} + \frac{p(\tilde{O}_{EST}|F0_{GT})}{p(\tilde{O}_{EST}|F0_{EST})}\right)$$

It can be interpreted that the error computes the likelihood that the period values generated by the two F0 values would lead to similar F0 estimation results. $E_{harm}$ is computed for each point on the trajectory and averaged to obtain cumulative measure. Figure 11 demonstrates the output of the measure for several estimated trajectories. For more details about the period likelihood function and motivation behind the circular von-Mises distribution, the reader is referred to [4].

### Optimal feature segregation method

If the acoustic signal contains a mixture of two voices, then the observation $O(n)$ can be segregated into *foreground observation* $O_{\mathrm{F}}(n)$ and *background observation* $O_{\mathrm{B}}(n)$. Following the assumption that each channel set represents only one voice, each set $G_{cn}$ is assigned to either the foreground or the background. We use the approach from [4] which proved to provide segregation sufficient to simultaneously track the fundamental frequency of 2 competing voices. The likelihood that sPAF belong to the foreground voice is compared with the likelihood that they belong to the background voice.

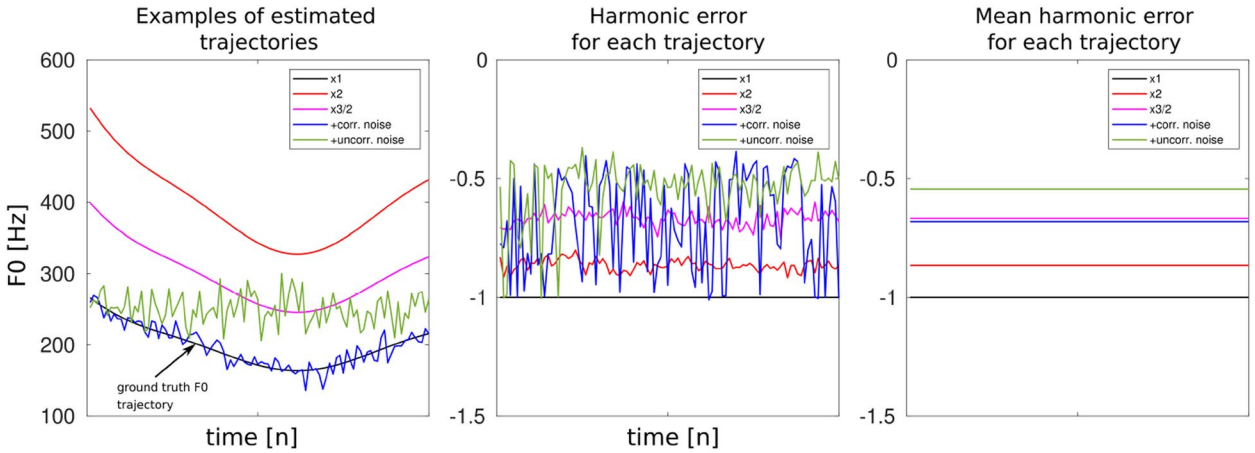


**Fig. 11** Examples of trajectories in a different relation to the ground truth trajectory and the corresponding harmonic errors. The lowest error is found for the ground truth (GT) trajectory itself (in black), next for 2× the GT trajectory (in red), GT trajectory with additional noise (in blue), $\frac{2}{3}\times$ the GT trajectory (in magenta), and the highest error is found for noise uncorrelated with the GT trajectory. Note that the error in terms of absolute deviation is the highest for 2× the GT trajectory (in red). The harmonic error is low because it is likely to obtain similar period values for those trajectories

Specifically, the integrated (across elements in the set and frequency channels) period likelihood given the true $F0$ of the first voice was compared with the integrated period likelihood given the true $F0$ of the second voice:

**FOR all** $G_{cn}$ in $O(n)$

*Compute period likelihood given foreground ground truth* $F0$ :

$$L_{\mathrm{F}} = \prod_{c}\sum_{m}\frac{1}{M_{cn}}p(P_{cnm}|F0_{\mathrm{F}}(n-1))$$

*Compute period likelihood given background ground truth* $F0$ :

$$L_{\mathrm{B}} = \prod_{c}\sum_{m}\frac{1}{M_{cn}}p(P_{cnm}|F0_{\mathrm{B}}(n-1))$$

**IF** $L_{\mathrm{F}} > L_{\mathrm{B}}$

assign $G_{cn}$ to $O_{\mathrm{F}}(n)$

**ELSE**

assign $G_{cn}$ to $O_{\mathrm{B}}(n)$

**END**

**END**

**Acknowledgements**
This work was supported by the University of Oldenburg, Department of Medical Physics and Acoustics, and by the DFG Cluster of Excellence EXC 1077/1 "Hearing4all".

**Authors' contributions**
Joanna Luberadzka: conceptualization, study design, software and simulations, data analysis, writing, review, and editing. Hendrik Kayser: supervision, conceptualization, study design, review, and editing. Jörg Lücke: supervision, conceptualization, review, and editing. Volker Hohmann: supervision, conceptualization, study design, review, and editing.

**Funding**
Open Access funding enabled and organized by Projekt DEAL. Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 352015383 – SFB 1330.

**Availability of data and materials**
The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Competing interests**
The authors declare that they have no competing interests.

Received: 22 November 2022 Accepted: 3 May 2024
Published online: 22 May 2024

**References**

1. M. Cooke, A glimpsing model of speech perception in noise. J. Acoust. Soc. Am. **119**(3), 1562–1573 (2006)
2. E. Schoenmaker, S. van de Par. Intelligibility for binaural speech with discarded low-SNR speech components, in *Physiology, psychoacoustics and cognition in normal and impaired hearing* (Springer International Publishing, 2016), pp. 73–81
3. R.L. Gregory, Perceptions as hypotheses. Philos. Trans. R. Soc. Lond. B Biol. Sci. **290**(1038), 181–197 (1980)
4. J. Luberadzka, H. Kayser, V. Hohmann, Making sense of periodicity glimpses in a prediction-update-loop–a computational model of attentive voice tracking. J Acoust. Soc. Am. **151**(2), 712–737 (2022)
5. K.J. Woods, J.H. McDermott, Attentive tracking of sound sources. Curr. Biol. **25**(17), 2238–2246 (2015)
6. A. Josupeit, N. Kopčo, V. Hohmann, Modeling of speech localization in a multi-talker mixture using periodicity and energy-based auditory features. J. Acoust. Soc. Am. **139**(5), 2911–2923 (2016)

7. A. Josupeit, V. Hohmann, Modeling speech localization, talker identification, and word recognition in a multi-talker setting. J. Acoust. Soc. Am. **142**(1), 35–54 (2017)
8. A. Josupeit, E. Schoenmaker, S. van de Par, V. Hohmann, Sparse periodicity-based auditory features explain human performance in a spatial multitalker auditory scene analysis task. Eur. J. NeuroSci. (2018)
9. M.S Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on signal processing. **50**(2), 174–88 (2002)
10. D. Van Ravenzwaaij, P. Cassey, S.D. Brown, A simple introduction to markov chain monte-carlo sampling. Psychon. Bull. Rev. **25**(1), 143–154 (2018)
11. D.F. Specht et al., A general regression neural network. IEEE Trans. Neural Netw. **2**(6), 568–576 (1991)
12. J. Luberadzka, H. Kayser, V. Hohmann. Estimating fundamental frequency and formants based on periodicity glimpses: A deep learning approach, in *2020 IEEE International Conference on Healthcare Informatics (ICHI)*, vol. 30 (IEEE, 2020), pp. 1–6
13. V. Hohmann. Method for extracting periodic signal components, and apparatus for this purpose (Google Patents, 2006). US Patent App. 11/223125
14. J. Luberadzka, H. Kayser, V. Hohmann, Glimpsed periodicity features and recursive Bayesian estimation for modeling attentive voice tracking. Universitätsbibliothek der RWTH Aachen; 2019.
15. Z. Chen et al., Bayesian filtering: From kalman filters to particle filters, and beyond. Stat. **182**(1), 1–69 (2003)
16. S. Ruder, An overview of gradient descent optimization algorithms (2016). arXiv preprint arXiv:1609.04747
17. D.H. Klatt, Software for a cascade/parallel formant synthesizer. J. Acoust. Soc. Am. **67**(3), 971–995 (1980)
18. J.G. Bernstein, A.J. Oxenham, Pitch discrimination of diotic and dichotic tone complexes: harmonic resolvability or harmonic number? J. Acoust. Soc. Am. **113**(6), 3323–3334 (2003)
19. S. Mittal, A. Lamb, A. Goyal, V. Voleti, M. Shanahan, G. Lajoie, M. Mozer, Y. Bengio. Learning to combine top-down and bottom-up signals in recurrent neural networks with attention over modules, in *International Conference on Machine Learning*, vol. 21 (PMLR, 2020), pp. 6972–6986
20. D. Husmeier, J.G. Taylor, Predicting conditional probability densities of stationary stochastic time series. Neural Netw. **10**(3), 479–497 (1997)
21. B. Lim, S. Zohren, Time-series forecasting with deep learning: a survey. Phil. Trans. R. Soc. A. **379**(2194), 20200209 (2021)
22. M.F. Stollenga, J. Masci, F. Gomez, J. Schmidhuber, Deep networks with internal selective attention through feedback connections. Advances in neural information processing systems 27, (2014)
23. D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate (2014). arXiv preprint arXiv:1409.0473
24. A. Rosenfeld, M. Biparva, J.K. Tsotsos. Priming neural networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2018), pp. 2011–2020
25. D.J. Rezende, S. Mohamed, D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models, in *International conference on machine learning* (PMLR, 2014), pp. 1278–1286
26. Kingma DP, Welling M. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114. (2013)
27. S. Ramchandran, G. Tikhonov, K. Kujanpää, M. Koskinen, H. Lähdesmäki. Longitudinal variational autoencoder, in *International Conference on Artificial Intelligence and Statistics* (PMLR, 2021), pp. 3898–3906
28. V. Fortuin, D. Baranchuk, G. Rätsch, S. Mandt. Gp-vae: Deep probabilistic time series imputation, in *International conference on artificial intelligence and statistics* (PMLR, 2020), pp. 1651–1661
29. M. Ashman, J. So, W. Tebbutt, V. Fortuin, M. Pearce, R.E. Turner, Sparse gaussian process variational autoencoders (2020). arXiv preprint arXiv:2010.10177
30. A. Nazabal, P.M. Olmos, Z. Ghahramani, I. Valera, Handling incomplete heterogeneous data using VAEs. Pattern Recognit. **107**, 107501 (2020)
31. D.S. Brungart, P.S. Chang, B.D. Simpson, D. Wang, Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. J. Acoust. Soc. Am. **120**(6), 4007–4018 (2006)
32. E.d. Boer, Pitch of inharmonic signals. Nat. **178**(4532), 535–536 (1956)
33. J.F. Schouten, R. Ritsma, B.L. Cardozo, Pitch of the residue. J. Acoust. Soc. Am. **34**(9B), 1418–1424 (1962)
34. P.A. Cariani, B. Delgutte, Neural correlates of the pitch of complex tones. ii. pitch shift, pitch ambiguity, phase invariance, pitch circularity, rate pitch, and the dominance region for pitch. J. Neurophys. **76**(3), 1717–1734 (1996)
35. E Terhardt. On the role of ambiguity of perceived pitch in music, in *Proc. 13th ICA Belgrade* (1989), pp. 35–38
36. P.F. Assmann, Q. Summerfield, Modeling the perception of concurrent vowels: vowels with different fundamental frequencies. J. Acoust. Soc. Am. **88**(2), 680–697 (1990)
37. M.R. Saddler, R. Gonzalez, J.H. McDermott, Deep neural network models reveal interplay of peripheral coding and stimulus statistics in pitch perception. Nature communications. **12**(1), 7278 (2021)


## Publisher's Note