CREPE: A Convolutional Representation of Pitch Estimation

Jong Wook Kim[1], *JustinSalamon*[1,2], *PeterLi*[1], *JuanPabloBello*[1]

Music and Audio Research Laboratory, New York University
Center for Urban Science and Progress, New York University

ICASSP 2018 - Proceedings

CSC2518, Sherry Wang

February 19, 2019

Overview

Auditory Attribute of Musical Tones

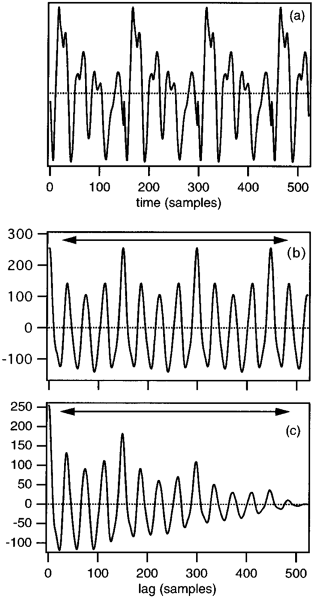
- Pitch ($\approx$ fundamental frequency/F0, for voiced speech)
- Timbre (spectrum, envelope, instruments)
- Duration (timing, pauses, rate)
- Loudness (amplitude/energy)

Pitch Estimation

Three mostly used methods for pitch estimation:

- Autocorrelation of Speech - used by baseline: pYIN
- Cepstrum Pitch Determination
- Single Inverse Filter Tracking(SIFT) Algorithm

Pitch Estimation by Autocorrelation Method



Autocorrelation Function

$$r_t(\tau) = \sum_{j=t+1}^{t+W} x_j x_{j+\tau} \qquad (1)$$

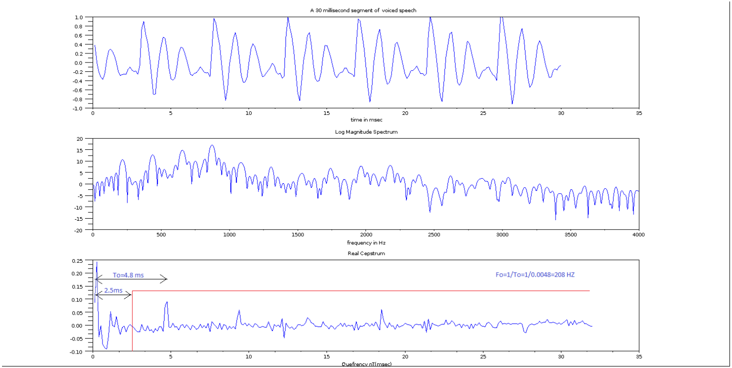
$$r'_t(\tau) = \sum_{j=t+1}^{t+W-\tau} x_j x_{j+\tau} \qquad (2)$$

The second largest peak location in samples gives T_0 and thus pitch is computed as:

$$F_0 = F_s / T_0 \qquad (3)$$

F_s: sampling frequency
T_0: pitch period in samples

Cepstrum Pitch Determination



The largest peak location gives T_0 and thus pitch is computed as:

$$F_0 = 1/T_0 \tag{4}$$

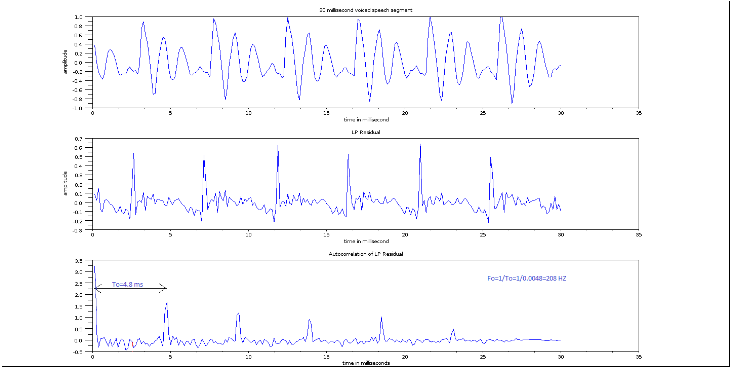
F_s: sampling frequency $\quad$ T_0: pitch period in time

Pitch estimation by SIFT method

Linear prediction (LP)

- A speech sample can be approximated as a linear combination of past samples.
- Obtain a unique set of predictor coefficients by minimizing the sum of the squared differences between the actual speech samples and the linearly predicted ones over a finite interval.
- Decomposes the speech into two highly independent components:
  1. Vocal tract parameters (LP coefficients)
  2. Glottal excitation (LP residual).
- The autocorrelation of LP residual will therefore have unambiguous peaks representing the pitch period 'T0' information.

Pitch estimation by SIFT method

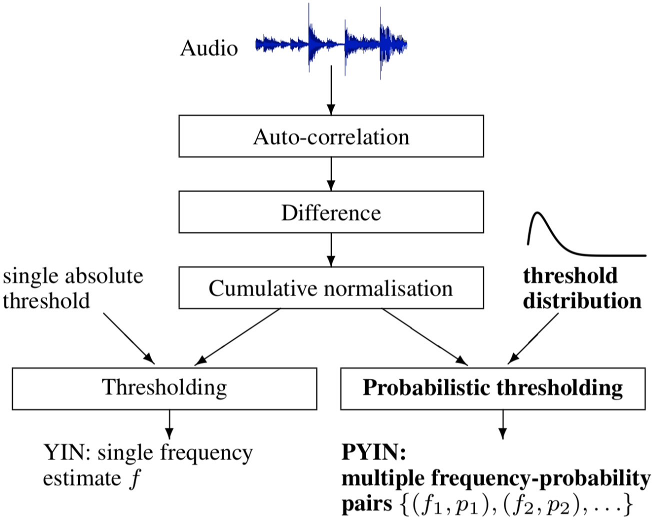


A single peak picking can be employed for the estimation of pitch period T_0 as illustrated in the figure. Pitch is computed as:

$$F_0 = 1/T_0 \tag{5}$$

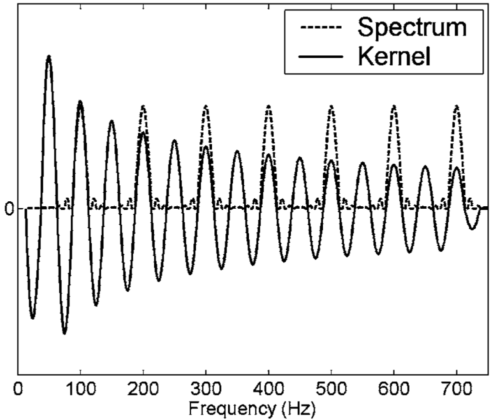
F_s: sampling frequency $\qquad$ T_0: pitch period in time

Baseline: YIN and pYIN



Baseline: SWIPE Pitch Estimation

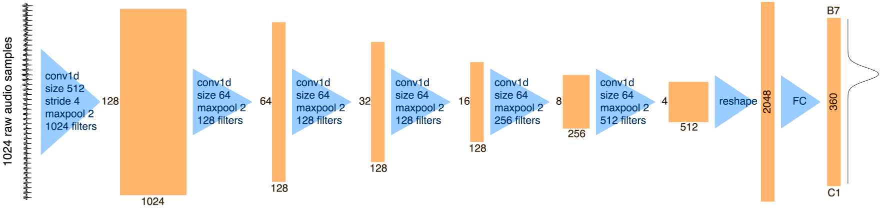
Sawtooth Waveform Inspired Pitch Estimator(SWIPE)



Shortcomings of existing methods and Motivation

- The development of well-performed systems solely depends on devising a robust **candidate-generating function**(i.e. heuristics) and/or **sophisticated post-processing steps**.
- **None** of the model **directly learn from data**, except for manual hyper-parameter tuning.
- In **other problems** in music information retrieval, e.g. chord ID, beat detection, **data-driven methods have been shown consistently out-perform heuristic approaches**.
- Current methods still produce noisy results for uncommon instruments and highly fluctuated pitch curves.

CREPE - Architecture



- Input: 1024 samples excerpt from time-domain audio signal, 16kHz sampling rate.
- 6 convolutional layers, resulting in 2048 latent representation
- Sigmoid activation, deterministic
- 360-dim output vector where each frequency bin covers 20 **cents**.

CREPE - Output Interpretation

Cent: A unit representing musical intervals relative to a reference pitch f_{ref} in Hz, defined as a function of frequency f in Hz:

$$c(f) = 1200 \times log_2 \frac{f}{f_{ref}} \tag{6}$$

Where $f_{ref} = 10Hz$ is used throughout the experiment. The 360 pitch values are denoted $c_1, c_2, ..., c_{360}$, covers **6 octaves** from **C1** to **B7**. Resulting pitch estimate $\hat{c}$ is the weighted average of the associated c_i.

$$\hat{c}(f) = \frac{\sum_{i=1}^{360} \hat{y}_i c_i}{\sum_{i=1}^{360} \hat{y}_i} \tag{7}$$

and obtain the estimated frequency:

$$\hat{f} = f_{ref} \times 2^{\hat{c}/1200} \tag{8}$$

CREPE - Datasets

RWC-syth

- **6.16 hours** of audio synthesized from the RWC Music Database.
- Have perfect control over the $F0$ of the resulting signal.
- Synthesized using a fixed sum of a small number of sinusoidal, highly homogeneous in timbre and represents an **over-simplified scenario.**

MDB-stem-synth

- 230 tracks with 25 instruments, totaling 15.56 hours of audio.
- Monophonic stems taken from MedleyDB and re-synthesized, with a **perfect f0 annotation** that **maintains the timbre and dynamics** of the original track.
- Representing a **real-world scenario**.

CREPE - Target Output and Training

Target Output
360-dimensional vector(same as model's output). Frequency bin with the ground truth frequency is given a magnitude of 1 and then Gaussian blurred.

$$y_i = exp(-\frac{(c_i - c_{true})^2}{2 \times 25^2}) \quad (9)$$

Cross entropy loss between predicted vector $\hat{y}$ and target vector y

$$L(y, \hat{y}) = \sum_{i=1}^{360}(-y_i log \hat{y}_i - (1 - y_i) log(1 - \hat{y}_i)) \quad (10)$$

- Optimized by ADAM optimizer with learning rate 0.0002.
- Trained for 32 epochs with 500 batches and batch size 32.
- Each convolutional layer is followed with a drop out layer with drop-out rate 0.25.

Experiment and Evaluation Criteria

Methodology

5-fold cross-validation, 60/20/20 train, validation and test split.

Evaluation

Raw Pitch Accuracy(RPA) and Raw Chorma Accuracy(RCA) within 50 cent(a quarter-tone) threshold of the ground truth.

Added Noise - Audio Degradation Toolbox(ADT)

4 Noise sources: pub, while, pink, brown

Use different Signal-to-Noise Ratios(SNR): $, 40, 30, 20, 10, 5, 0dB$

Results - Pitch Accuracy with 50 cents(standard) threshold

Table 1. Average raw pitch/chroma accuracies and their standard deviations, tested with the 50 cents threshold

Dataset	Metric	CREPE	pYIN	SWIPE
RWC-synth	RPA	**0.999±0.002**	0.990±0.006	0.963±0.023
	RCA	**0.999±0.002**	0.990±0.006	0.966±0.020
MDB-stem-synth	RPA	**0.967±0.091**	0.919±0.129	0.925±0.116
	RCA	**0.970±0.084**	0.936±0.092	0.936±0.100

Results - Pitch Accuracy with different evaluation thresholds

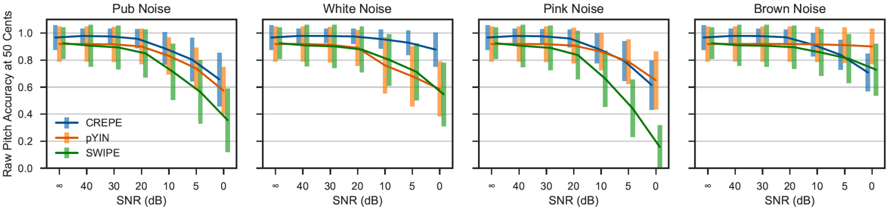
Table 2: Average raw pitch accuracies and their standard deviations, with different evaluation thresholds.

Dataset	Threshold	CREPE	pYIN	SWIPE
RWC-synth	50 cents	**0.999±0.002**	0.990±0.006	0.963±0.023
	25 cents	**0.999±0.003**	0.972±0.012	0.949±0.026
	10 cents	**0.995±0.004**	0.908±0.032	0.833±0.055
MDB-stem-synth	50 cents	**0.967±0.091**	0.919±0.129	0.925±0.116
	25 cents	**0.953±0.103**	0.890±0.134	0.897±0.127
	10 cents	**0.909±0.126**	0.826±0.150	0.816±0.165

This suggests that CREPE is especially preferable when even minor deviations from the true pitch should be avoided as best as possible.

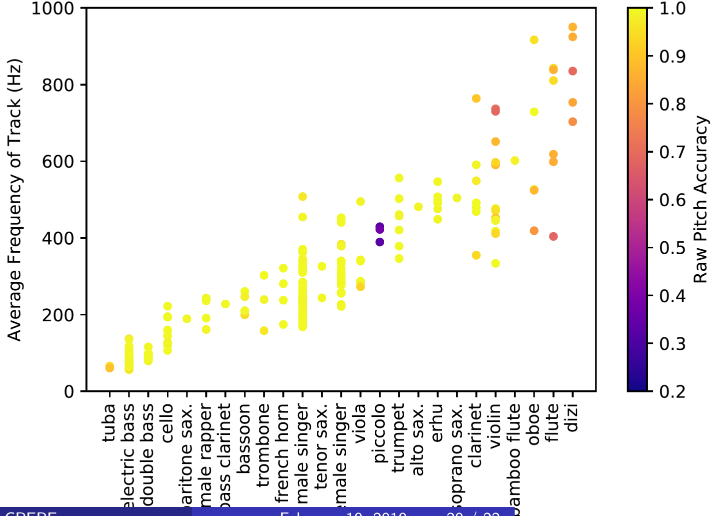
Results - Niose Robustness

Pitch tracking performance when additive noise signals.
The error bars are centered at the average raw pitch accuracies and span the first standard deviations.
With brown noise being a notable exception, CREPE shows the highest noise robustness in general.



Results - Performance by Instrument

RPA of CREPEs predictions on each of the 230 tracks in MDB-stem-synth with respect to the instrument, sorted by the average frequency.



Conclusion

Contribution of this model

1. State-of-the-art performance on both datasets with homogeneous and heterogeneous timbre.
2. Highly accurate even at strict evaluation threshold.
3. More robust to added noise.
4. Innovative data-driven pitch tracking algorithm.

Future Work

1. Model should be invariant to all transformations that do not effect pitch. Use data augmentation to generate transformed and degraded signal and make the model to learn the invariance.
2. Robustness can be improve by applying pitch-shift to cover a wider pitch range.
3. Enforcing temporal smoothness to improve the performance, by using CRNN.

Q & A