Single Pitch Detection

(Thanks to Prof Zhiyao Duan of U. Rochester for the slides on YIN)

What is pitch?

- A perceptual attribute, so it is subjective
- Only defined for (quasi) harmonic sounds
  - Harmonic sounds are periodic, and the period is 1/F0.
- Can be reliably matched to fundamental frequency (F0)
  - In computer audition, people do not often discriminate pitch from F0
- F0 is a physical attribute, so it is objective

Why is pitch detection important?

- Harmonic sounds are ubiquitous
  - Music, speech, bird singing

- Pitch (F0) is an important attribute of harmonic sounds, and it relates to other properties
  - key, scale (e.g., major, minor), melody, emotion, etc.
  - Speech intonation → word disambiguation (for tonal language), statement/question, emotion, etc.

What emotion?

General Process of Pitch Detection

- Segment audio into time frames
  - Pitch changes over time
- Detect pitch (if any) in each frame
  - Need to detect if the frame contains pitch or not
- Post-processing to consider contextual info
  - Pitch contours are often continuous

How long should the frame be?

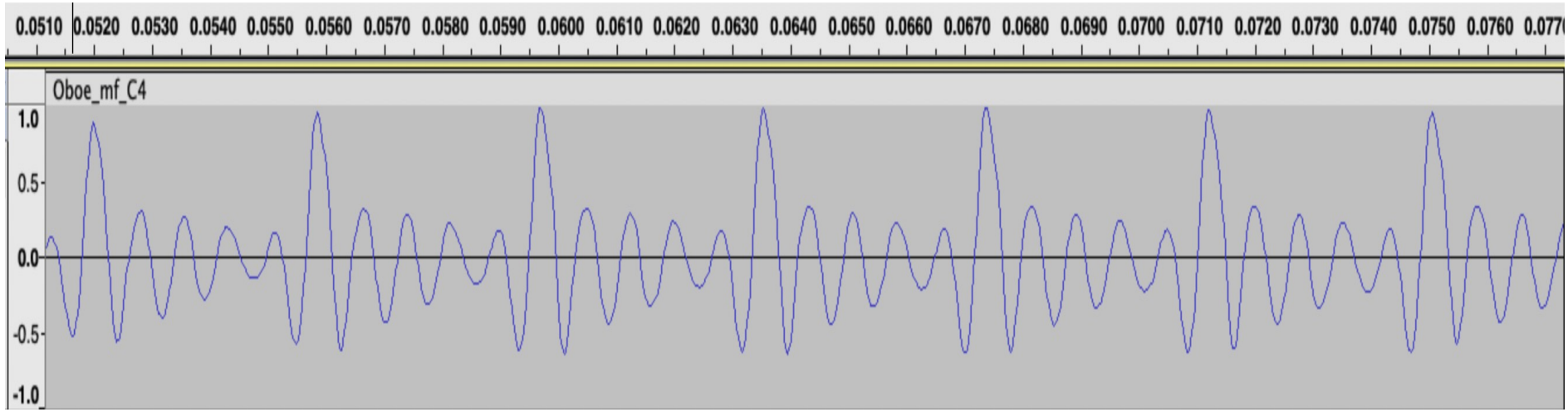
- Too long:

  Contains multiple pitches (low time resolution)

- Too short

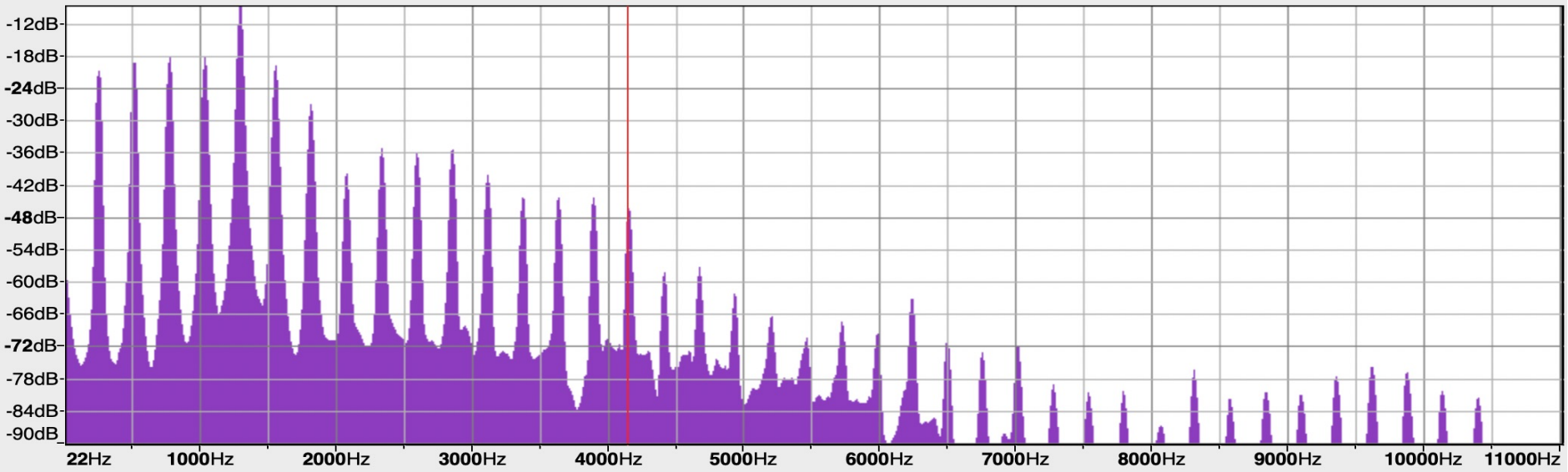
  Can't obtain reliable detection (low freq resolution)

  Should be at least 3 periods of the signal



For speech or music, how long should the frame be?

Spectrum of the same oboe C4



- Spectral peaks have harmonic relations.
  F0 is the greatest common divisor

- Spectral peaks are equally spaced.
  F0 is the frequency gap

Pitch Detection Cues

- Time domain signal is periodic.
  - F0 = 1/period
- Spectral peaks have harmonic relations.
  - F0 is the greatest common divisor
- Spectral peaks are equally spaced.
  - F0 is the frequency gap

- Time domain
  - Detect period
- Frequency domain
  - Detect the divisor
- Cepstrum domain
  - Detect the spacing

THE YIN PITCH DETECTOR

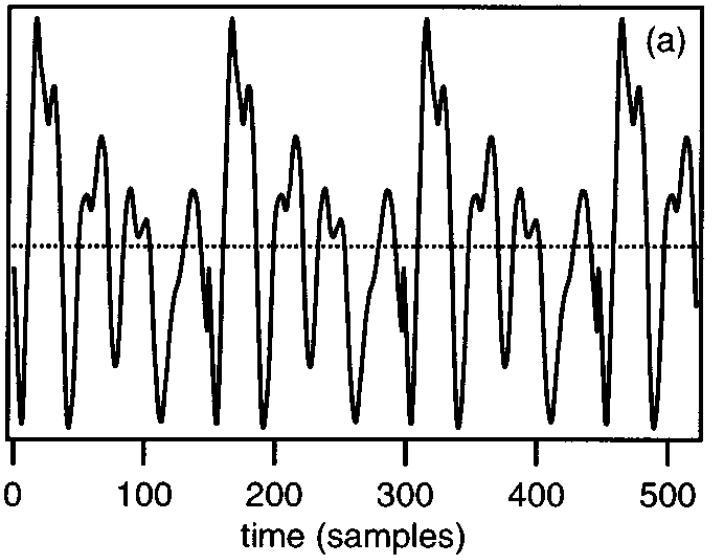
De Cheveigné, Alain, and Hideki Kawahara. "YIN, a fundamental frequency estimator for speech and music." *The Journal of the Acoustical Society of America* 111.4 (2002): 1917-1930.

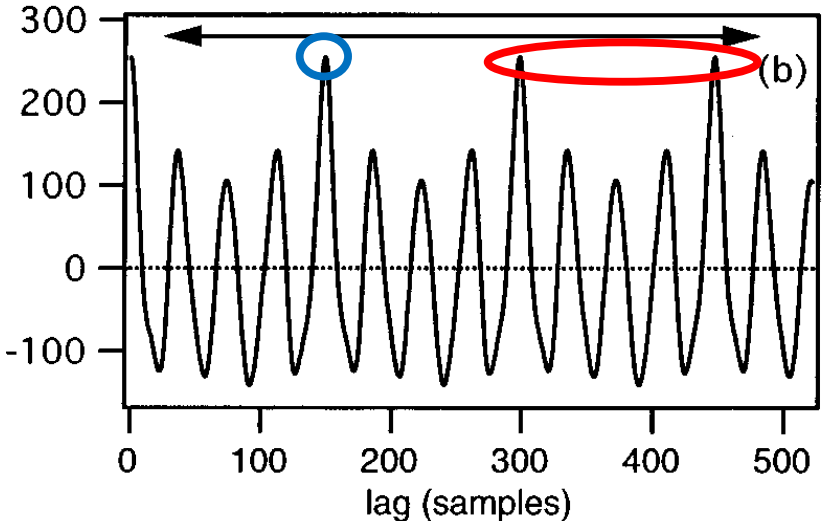
Time Domain: Autocorrelation

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

- A periodic signal correlates strongly with itself when offset by the period (and multiple periods)

- Problem: sensitive to peak amplitude changes
  - Which peak would be higher if signal amplitude increases?
  - Lower octave error (or sub-harmonic error)





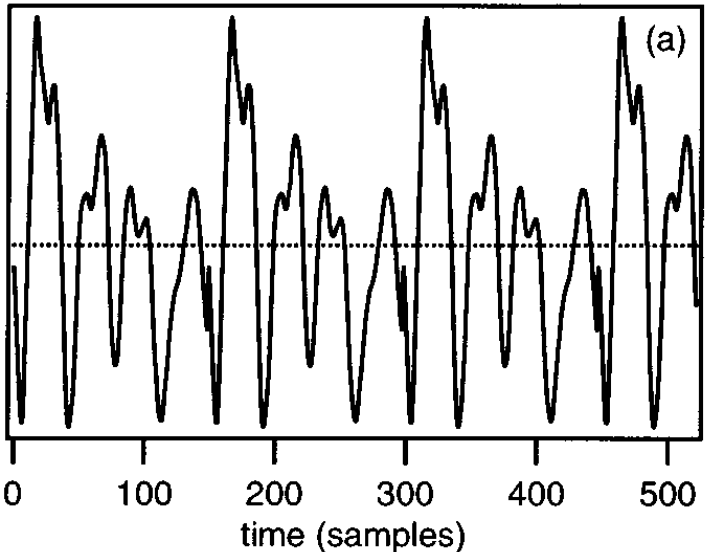
YIN – Step 2

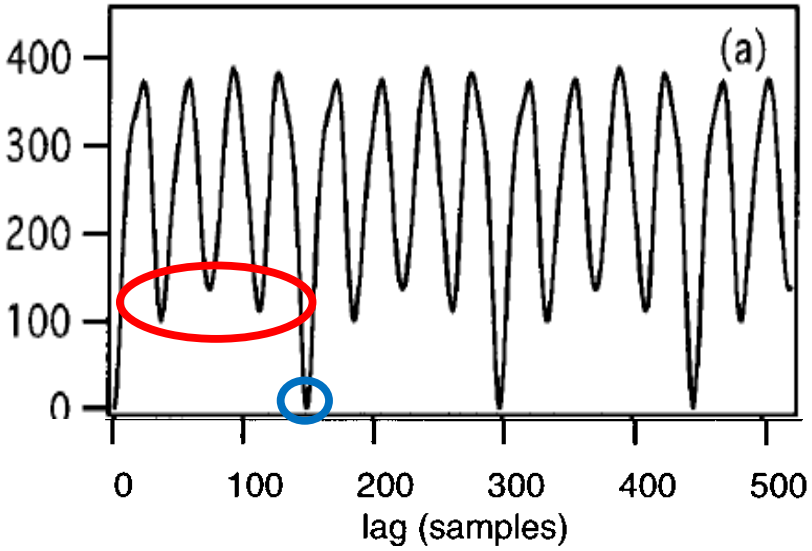
- Replace ACF with difference function

$$d_t(\tau) = \sum_{j=1}^{W} (x_j - x_{j+\tau})^2$$

- Look for dips instead of peaks, which is why it's called YIN opposed to YANG.
- Immune to amplitude changes
- Problem
  - Some dips close to 0 lag might be deeper due to imperfect periodicity

[de Cheveigne, 2002]

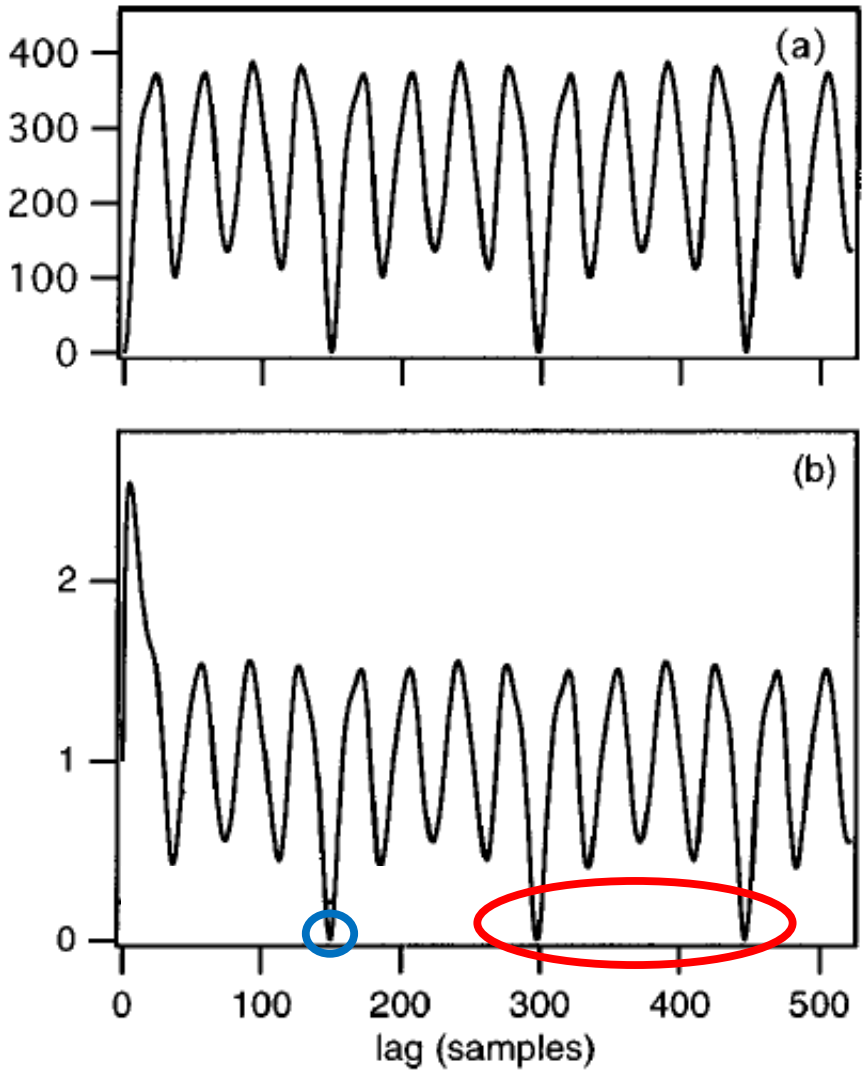




YIN – Step 3

- Cumulative mean normalized difference function

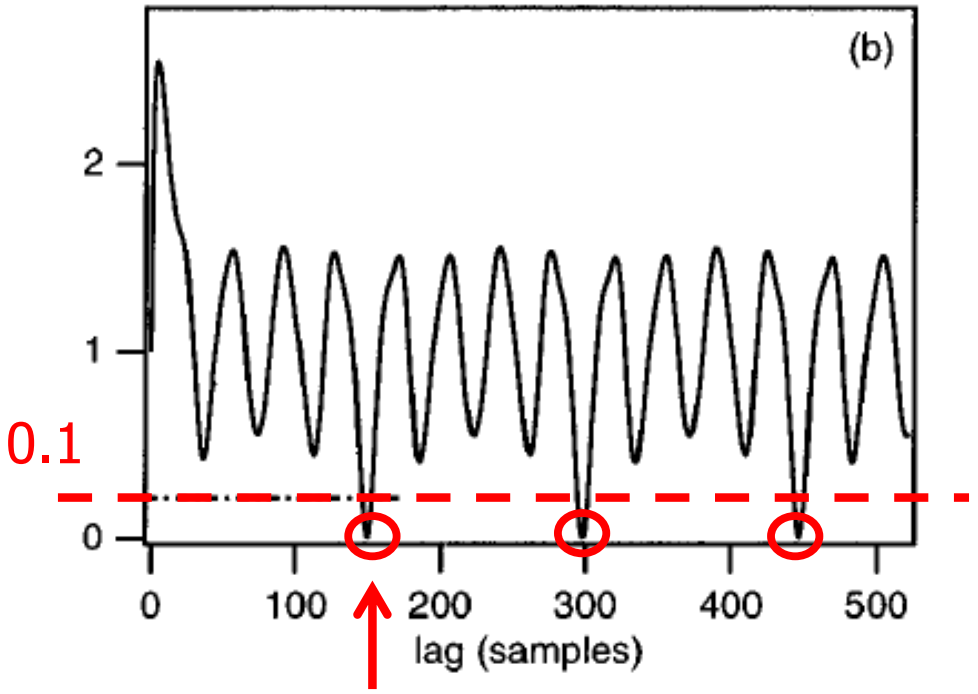
$$d'_t(\tau) = \begin{cases} 1, & \text{if } \tau = 0, \\ d_t(\tau) \Big/ \left[(1/\tau) \sum_{j=1}^{\tau} d_t(j) \right] & \text{other} \end{cases}$$



- Then take the deepest dip?
- Problem
  - May choose longer lag dips → lower octave error (or sub-harmonic error)

YIN – Step 4

- Absolute Threshold
  - Set threshold to say 0.1
  - Pick the first dip that exceeds the threshold



YIN – Step 5 & 6

- Step 5: parabolic interpolation to find the exact dip location
  - The dip location in the discrete world may deviate from the exact dip location
- Step 6: use the best local estimate
  - Some analysis points may be better than others (result in smaller d')
  - Use the pitch estimate from the best analysis point within the frame

Pitched or Non-pitched?

- Some frames may be silent or inharmonic, so they may not contain a pitch at all.

  Silence can be detected by RMS value.

  How about inharmonic frames?

- YIN: threshold on dip, aperiodicity

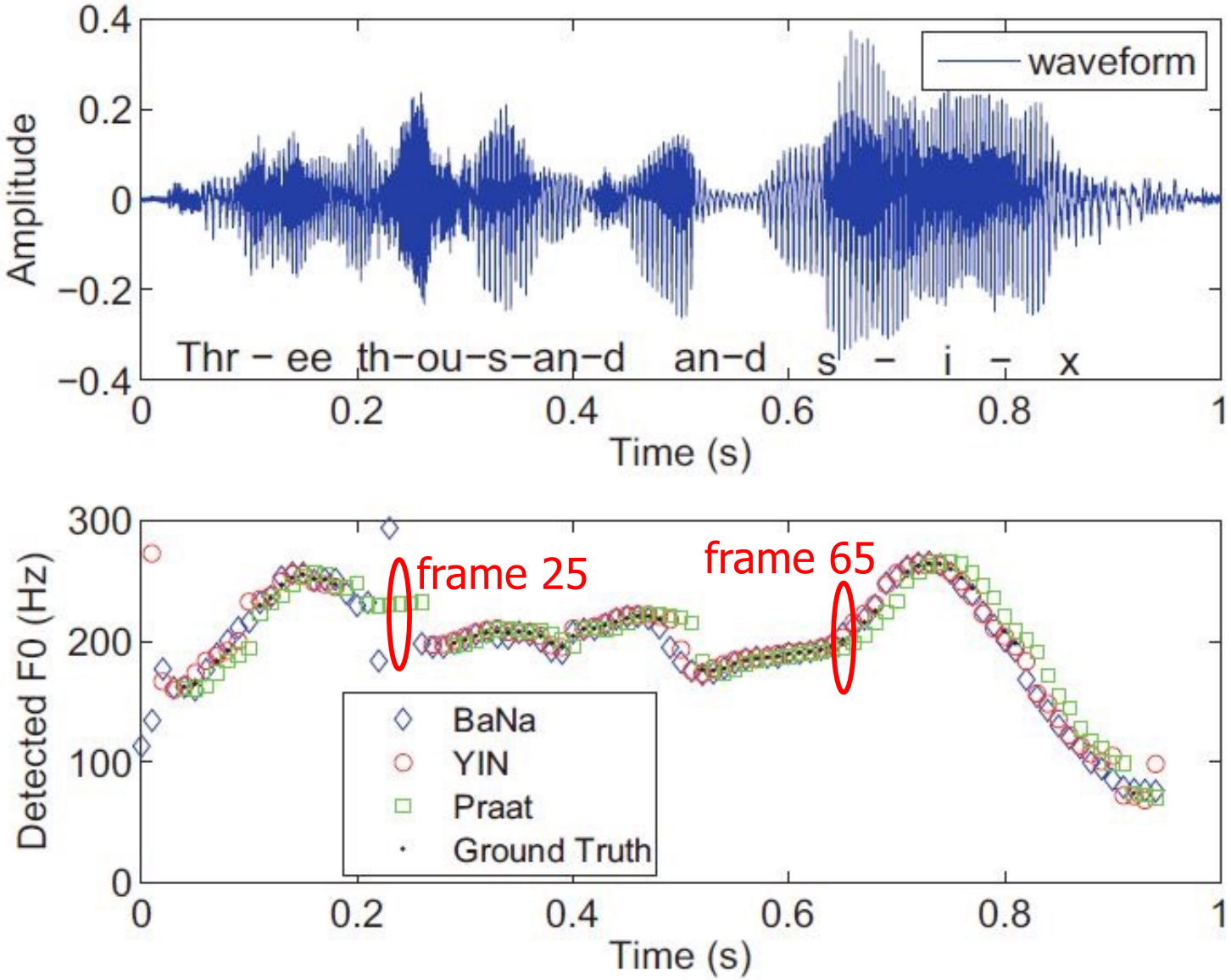
Evaluating performance

- What should we measure?
- Pitched/non-pitched classification error
- The difference between estimated pitch and true pitch
  - Threshold for speech: maybe 5% in Hz
  - Threshold for music: 1 quarter-tone (about 3% in Hz)
- The differences between estimated and true chroma
  - Helps distinguish pitch class error from octave error.

How do we get ground truth?

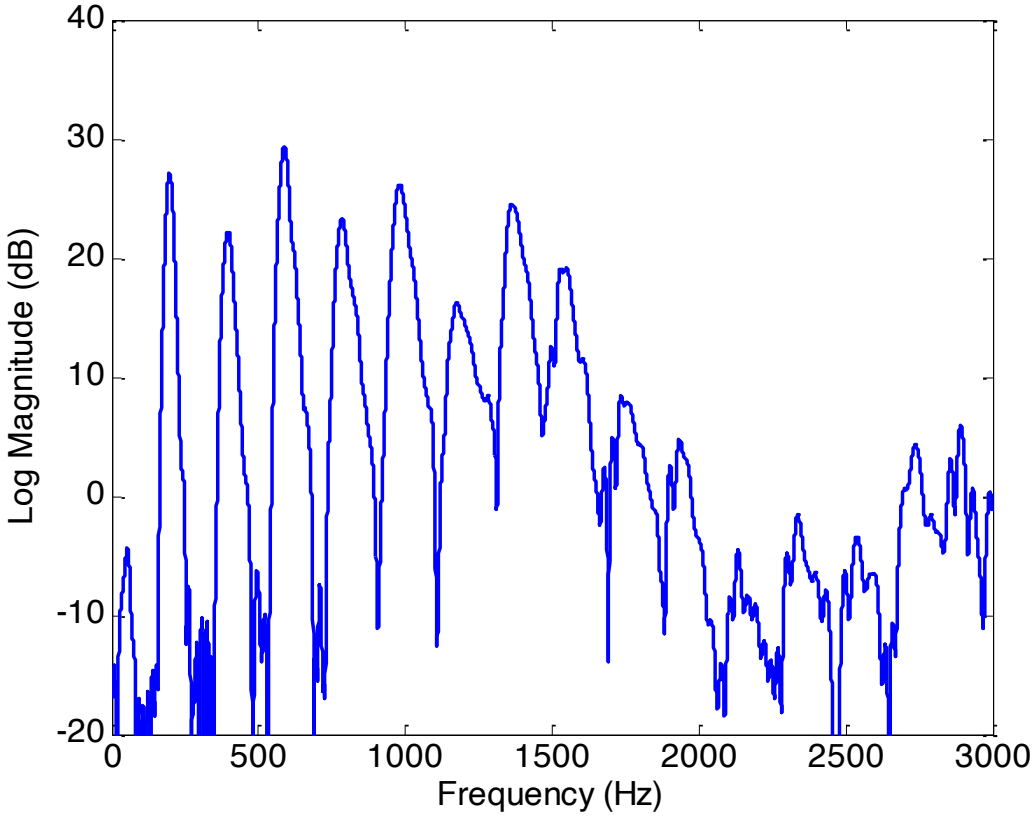
- Listen to the signal and inspect the spectrum to manually annotate (time consuming!)
- Automatic annotation using simultaneously recorded laryngograph signals for speech (not quite reliable!)
- Generate synthetic signals where you know the pitch (maybe not realistic!)
- Get creative? How can we use resampling to create data we can evaluate?

Different Methods vs. Ground-truth



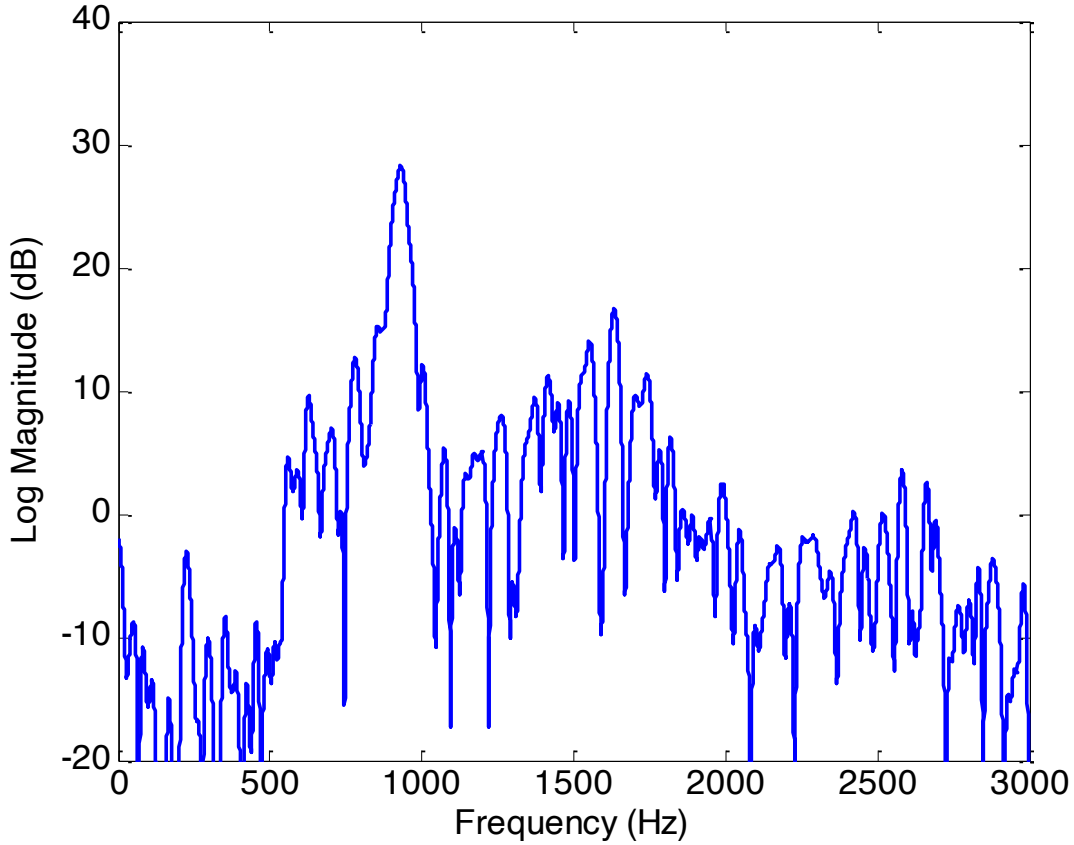
Spectrum of a voiced frame

- Has clear harmonic patterns
- Different methods give close results, and consistent to the ground-truth 196 Hz.



Spectrum of an unvoiced frame

- No clear harmonic patterns
- Different methods give inconsistent results.



Pitch Detection with Noise

- Can we still hear pitch if there is some background noise, say in a restaurant?

Violin + babble noise

- Will pitch detection algorithms still work?

- How can we improve pitch detection in noisy environments?

THE CREPE PITCH DETECTOR

CREPE: A Convolutional Representation for Pitch Estimation
Jong Wook Kim, Justin Salamon, Peter Li, Juan Pablo Bello.
Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2018.

Pitch detection in the 2020s

- People still use Yin a lot
- More people use CREPE
  - A convolutional neural network
  - Trained to output one pitch per time-frame
  - Millions of downloads
- …and CREPE descendants

CREPE architecture

Takes one frame of 1024 samples as input

Trained on 16kHz audio

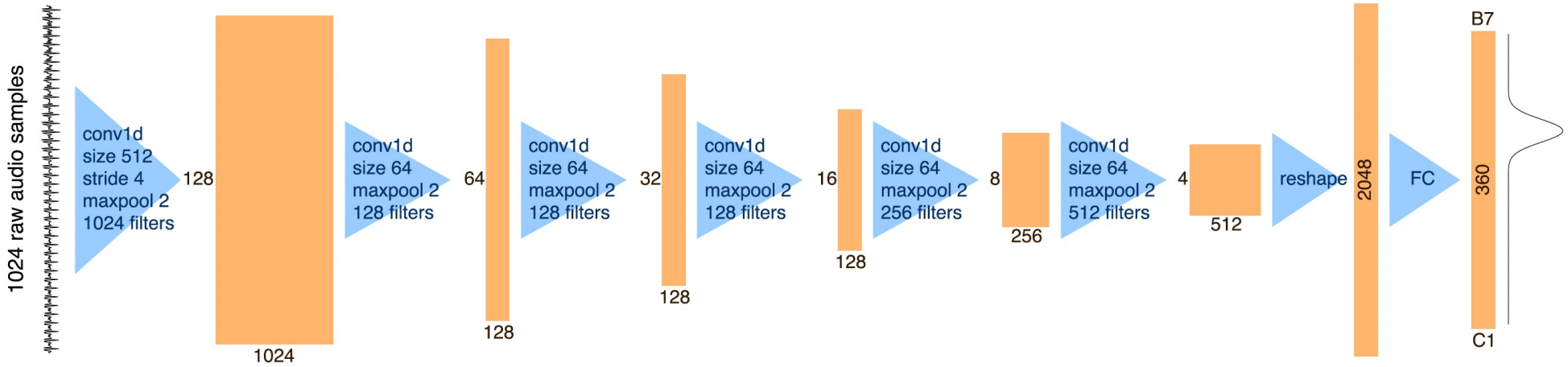
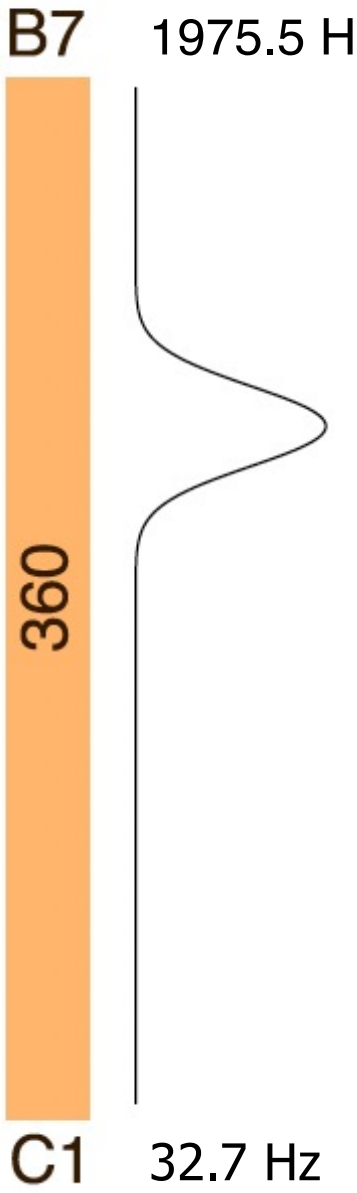


Fig. 1: The architecture of the CREPE pitch tracker. The six convolutional layers operate directly on the time-domain audio signal, producing an output vector that approximates a Gaussian curve as in Equation 3, which is then used to derive the exact pitch estimate as in Equation 2.

Last layer uses Sigmoid activations (ranging 0 to 1)

All convolutional layers use ReLU activations (0 to infinity)

CREPE output



Outputs are spaced at 20 cent intervals (1/5 of a musical half step).

There are 360 outputs, spanning 6 octaves.

Call this 360-element output vector $\hat{y}_i$

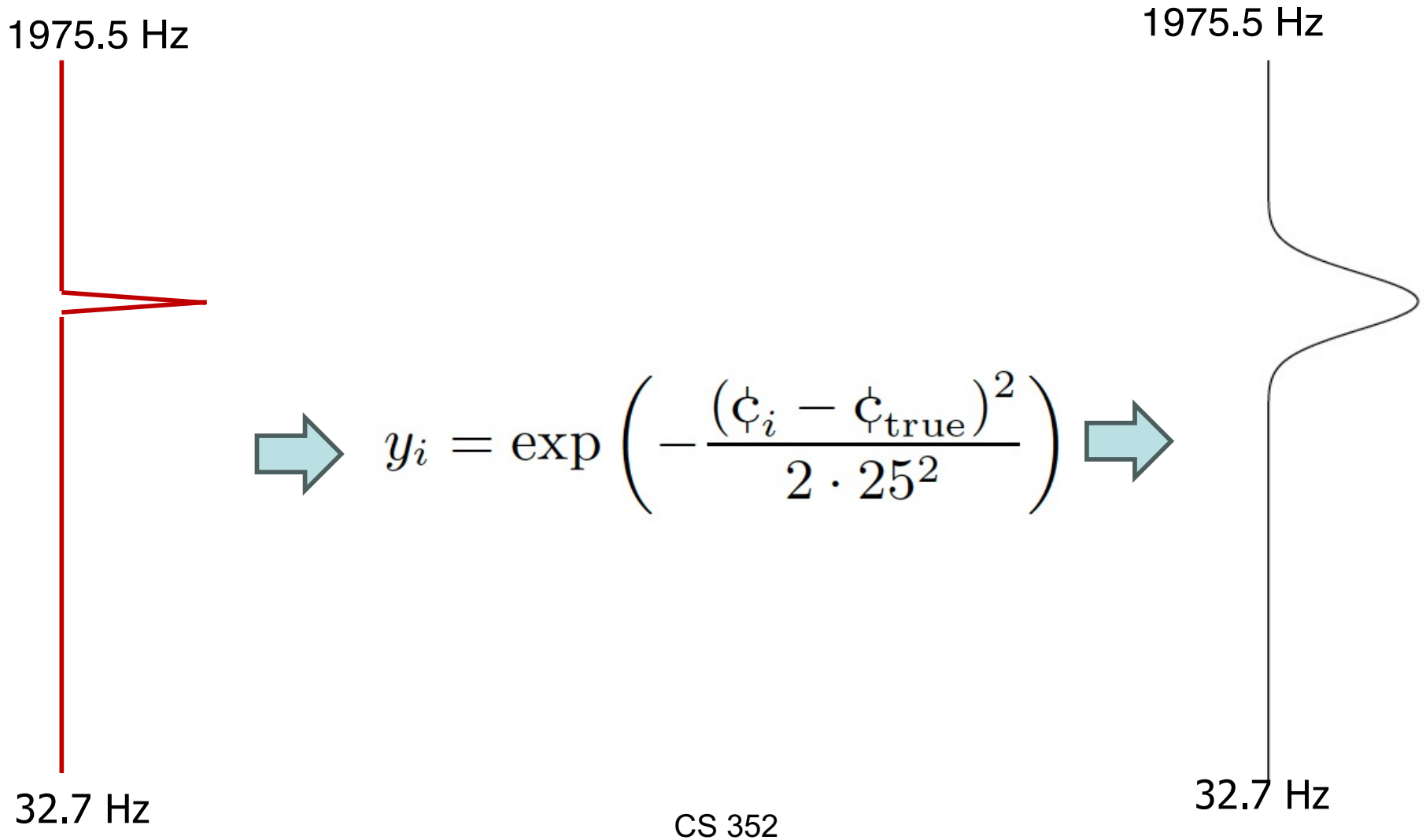
The final output of the model, $\hat{¢}$, is calculated by:

$$\hat{¢} = \frac{\sum_{i=1}^{360} \hat{y}_i ¢_i}{\sum_{i=1}^{360} \hat{y}_i}$$

CREPE ground-truth

Standard one-hot ground-truth:
1 at the right output. 0 elsewhere

What they use to train CREPE:
One-hot blurred by a Gaussian



Binary Cross Entropy loss function

The vector of outputs from crepe

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{i=1}^{360} (-y_i \log \hat{y_i} - (1 - y_i) \log(1 - \hat{y_i}))$$

The gaussian-blurred ground truth vector

Training data: MDB-stem-synth

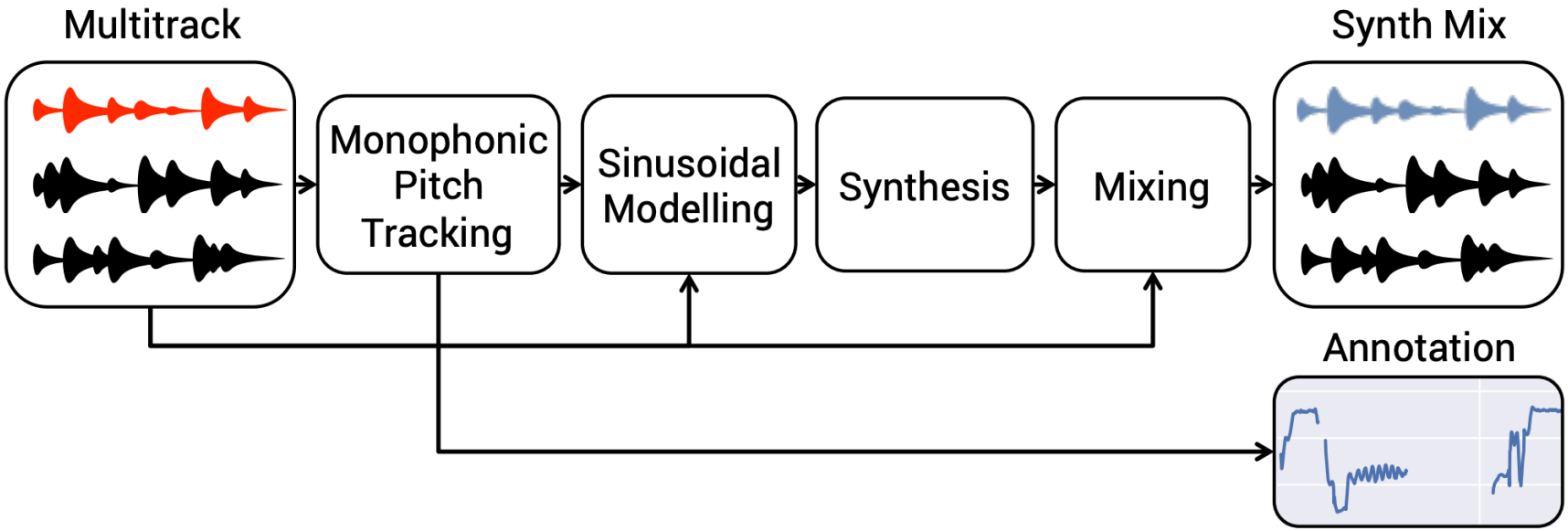
MDB-stem-synth is available here:
https://zenodo.org/records/1481172

15.5 hours of data, composed of 230 solo stems (tracks) from the MedlyDB dataset.

.

Making ground truth & training data

1. Take existing one-instrument audio tracks (MedlyDB dataset)
2. Pitch track them with an automatic pitch tracker (e.g. YIN)
3. Resynthesize the tracks using the pitch track to control pitch.
4. VOILA! You have perfectly pitch-tracked audio



How accurate are they?

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

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