CREPE:
A Convolutional Representation for Pitch Estimation

ICASSP 2018 Lecture Session AASP-L4.3: Music Signal Analysis and Processing
April 19, 2018

Jong Wook Kim, Justin Salamon, Peter Li, Juan Pablo Bello
Music and Audio Research Laboratory, New York University
Task: Monophonic Pitch Estimation

- A long-standing topic in audio signal processing
- A fundamental problem in music information retrieval
- With many applications
  - In a melody extraction system \( \text{Bosch and Gómez 2014} \)
  - Annotating multi-track datasets \( \text{Salamon et al. 2017} \)
  - Analyzing intonations in speech analysis
Background on Monophonic Pitch Estimation

A History of Engineering Heuristic Feature Extractor Functions

• Frequency-domain methods
  - Cepstrum¹ \(^{Noll 1967}\), SWIPE² \(^{Camacho and Harris 2008}\)

• Time-domain methods
  - \(f_{ACF}(\tau) = \sum x_t x_{t+\tau}\), \(f_{AMDF}(\tau) = \sum |x_t - x_\tau|\), \(f_{ASDF}(\tau) = \sum (x_t - x_{t-\tau})^2\)
  - \(YIN^{De Cheveigné and Kawahara 2002}\): cumulative mean normalized difference function
  - \(pYIN^{Mauch and Dixon 2014}\): a probabilistic extension to YIN – the state of the art

• These are all based on hand-crafted features and heuristics
Motivation

- Reported near-perfect accuracies are based on simplistic datasets
- We encountered many cases where the SoTA doesn’t do well: original pYIN
- Should benefit from data-driven methods, just like many other MIR tasks
Problem Formulation

1024-sample segment from 16,000 Hz audio

↓

Model

Gaussian curve

↓

360-D activation over 6 octaves, 20-cents-wide bins
Deep Model Architecture

![Diagram of a deep model architecture with convolutional layers and pooling operations. The architecture includes multiple convolutional layers with increasing filter sizes and pooling layers to reduce the dimensionality of the feature maps. The final layers include a fully connected (FC) layer and a softmax classifier.]
Post Processing and Optimization

• The model produces a 360-D activation vector for each input frame: Bittner et al. 2017

• Estimated pitch is then given as the (local) weighted average of the weights

\[ \hat{\chi} = \frac{\sum_{i=1}^{360} \hat{y}_i \hat{\chi}_i}{\sum_{i=1}^{360} \hat{y}_i} \]

• Optimization target: minimize the binary cross entropy:

\[ \mathcal{L}(y, \hat{y}) = \sum_{i=1}^{360} \left( -y_i \log \hat{y}_i - (1 - y_i) \log (1 - \hat{y}_i) \right) \]
Datasets and Evaluation

- For objective evaluation, we need a dataset with perfect pitch annotations.
- The datasets:
  - **RWC-synth** \textsuperscript{Mauch and Dixon 2014}: 6.16h, one timbre, on which pYIN was evaluated.
  - **MDB-stem-synth** \textsuperscript{Salamon et al. 2017}: 15.36h, 25 instruments from MedleyDB.
  - Listen: RWC-synth 1, RWC-synth 2, MDB-stem-synth 1, MDB-stem-synth 2.
- 5-fold cross validation and artist-conditional splits.
- Reporting pitch accuracies using \texttt{mir_eval} \textsuperscript{Raffel et al. 2014}:
  - Raw Pitch Accuracy (RPA)
  - Raw Chroma Accuracy (RCA)
Results: Pitch and Chroma Accuracy on RWC-synth

<table>
<thead>
<tr>
<th>Method</th>
<th>Raw Pitch Accuracy at 50 Cents</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREPE</td>
<td>99.98%</td>
</tr>
<tr>
<td>pYIN</td>
<td>99.19%</td>
</tr>
<tr>
<td>SWIPE</td>
<td>96.99%</td>
</tr>
</tbody>
</table>

RWC-synth
Results: Pitch and Chroma Accuracy on MDB-stem-synth

 MDB-stem-synth

CREPE: 99.40%
pYIN: 95.59%
SWIPE: 96.26%

Raw Pitch Accuracy at 50 Cents
Results: Thresholds

- **RWC-synth**
  - 10 cents
  - 25 cents
  - 50 cents
  - Thresholds: 0%, 90%, 99%, 99.9%, 100%
  - Raw Pitch Accuracy:
    - SWIPE
    - pYIN
    - CREPE

- **MDB-stem-synth**
  - 10 cents
  - 25 cents
  - 50 cents
  - Thresholds: 0%, 90%, 99%, 99.9%, 100%
  - Raw Pitch Accuracy:
    - SWIPE
    - pYIN
    - CREPE
Results: Noise Robustness

- **Pub Noise**
- **White Noise**
- **Pink Noise**
- **Brown Noise**

Raw Pitch Accuracy at 50 Cents

SNR (dB)

- CREPE
- pYIN
- SWIPE
Results: First-Layer Filters

- The filters \textbf{adapt} to the timbre distribution of the dataset
The Generalization Problem

- When the model trained on MDB-stem-synth is tested on my voice:
Fix 1: Argmax-Local Averaging

average?
Fix 1: Argmax-Local Averaging

- Local weighted average around the highest activation:

\[
\hat{c} = \frac{\sum_{i=m-4}^{m+4} \hat{y}_i \hat{c}_i}{\sum_{i=m-4}^{m+4} \hat{y}_i}, \quad m = \text{argmax}_i \hat{y}_i
\]
Fix 2: Train with ALL THE DATA!

- MIR-1K, Bach10, RWC-synth, MedleyDB, MDB-stem-synth, NSynth
The Pre-trained Model Release

- Fixed the generalization problem:

  - The highest activation is a good heuristic for voice activity detection (VAD)
  - An interactive demo: https://marl.github.io/crepe/
Summary

- Presented a data-driven neural network model as a state of the art method
  - Runs directly on time-domain audio signal
  - Robust with heterogeneous timbre and additive noise
  - Stays highly accurate, even with 10 cents threshold

- Try it today!
  
  $ pip install tensorflow  # or tensorflow-gpu
  $ pip install crepe  # install the CREPE package
  $ crepe audio.wav  # run pitch estimation on audio.wav
References


