A General-Purpose 32 ms Prosodic Vector for Hidden Markov Modeling

Kornel Laskowski[1,2], Mattias Heldner[3] & Jens Edlund[3]

[1]Carnegie Mellon University, Pittsburgh PA, USA
[2]Universität Karlsruhe, Karlsruhe, Germany
[3]KTH — Royal Institute of Technology, Stockholm, Sweden

8 September, 2008

Imagine you had ...

- a local representation of tone
  - estimated from a **single** ASR-size analysis frame
- which would **not** require:
  - prior determination of voicing
  - speaker normalization
- with separable codeword clusters for
  - absence of voicing
  - presence of voicing, constant F_0
  - presence of voicing, falling F_0, with rate of change
  - presence of voicing, rising F_0, with rate of change

Then you could do lots of things cheaply ...

Examples include:

- online prosodic modeling
- improved ASR for tonal languages
- enriched ASR for other languages
  - contrastive phone models
  - variously accented same-word lexicon entries
  - (word-conditioned) prosodic phrasing for free

Instead, currently you need to ...

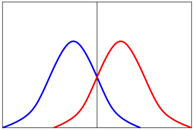
1. **run a pitch tracker**, which
   1. computes a local estimate of voicing and of pitch
   2. applies dynamic programming over a long observation time
2. **heuristically correct its output**, by
   1. pruning outliers, based on long-observation-time trends, and/or
   2. applying a piecewise linear approximation
3. **normalize for the speaker**, by
   1. determining a long-observation-time speaker norm
   2. applying the normalization to each frame
4. **treat unvoiced regions** by
   - interpolating inside them, or
   - posting exceptions in downstream modeling/handling
5. **compute a first-order log-difference**

What we will present ...

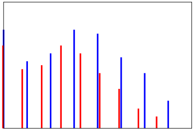
1. Fundamental Frequency Variation (FFV)
2. Applicability of the FFV Representation
   - speaker change prediction
   - speaker classification
   - dialog act classification
3. Several Basic Questions
   - feature transformation
   - feature regularization
   - concatenation with other features
   - runtime improvements
   - acoustic model complexity
4. Summary

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



time domain

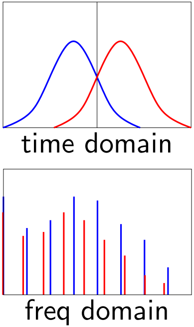


freq domain

- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

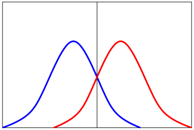
- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values

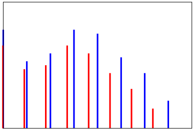


- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values

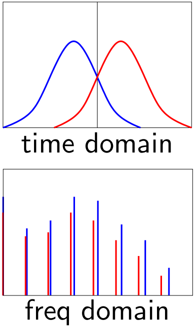

time domain


freq domain

- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

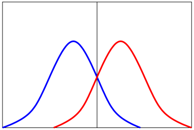
- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



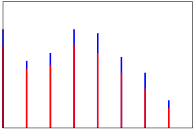
- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



time domain

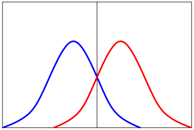


freq domain

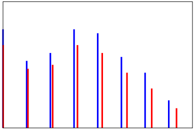
- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}\left[\rho\right]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



time domain

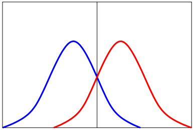


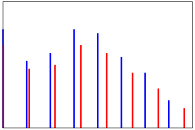
freq domain

- pass $\mathbf{g}\left(\rho\right)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values

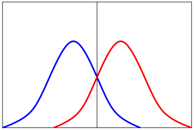




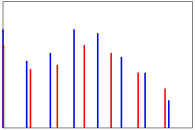
- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



time domain

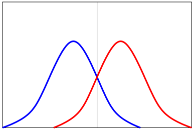


freq domain

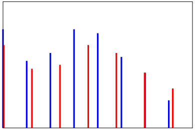
- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



time domain

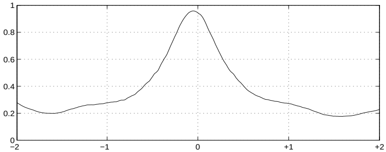


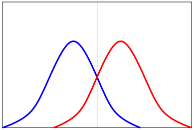
freq domain

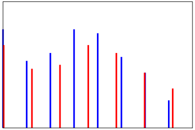
- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values



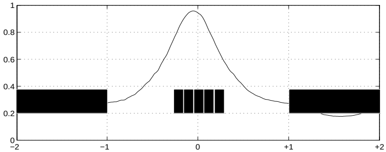


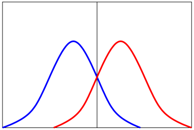


- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

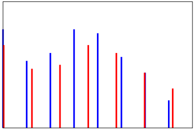
Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}[\rho]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values





time domain

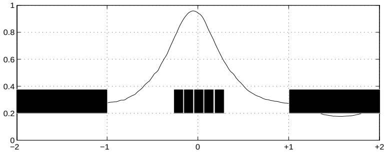


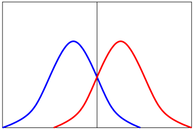
freq domain

- pass $\mathbf{g}(\rho)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

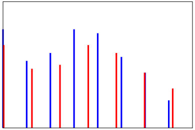
Computation, for Each (32 ms) Analysis Frame

- estimate the FFV spectrum $\mathbf{g}\left[\rho\right]$
  - estimate the power spectra $\mathbf{F}_L$ and $\mathbf{F}_R$
  - dilate $\mathbf{F}_R$ by a factor 2^ρ, $\rho > 0$
  - dot product with undilated $\mathbf{F}_L$
  - repeat for a continuum of ρ values





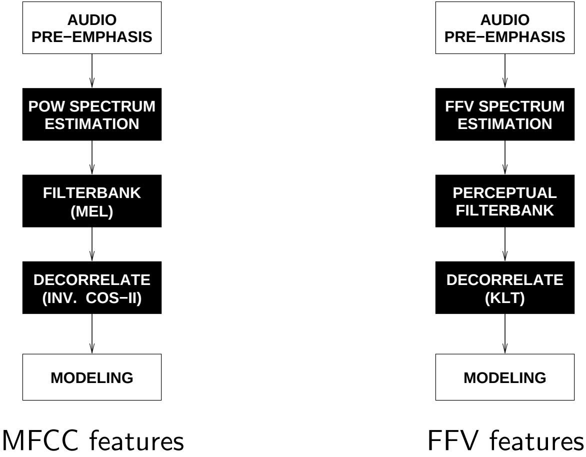
time domain



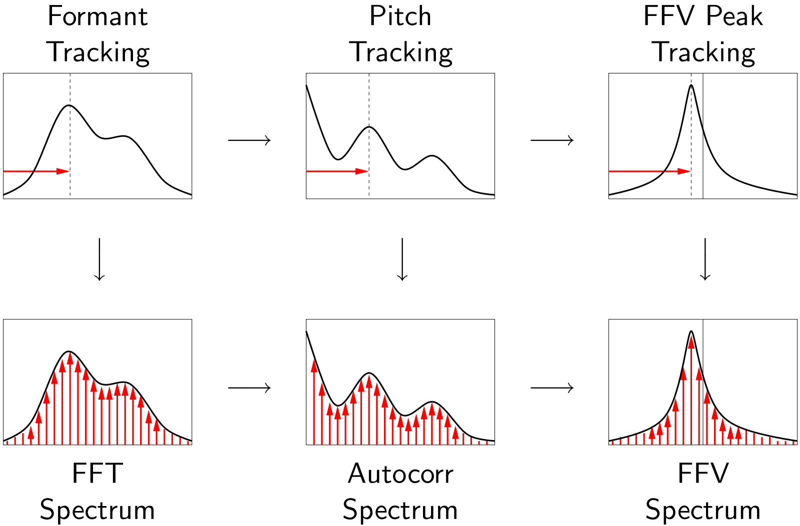
freq domain

- pass $\mathbf{g}\left(\rho\right)$ through a filterbank to yield $\mathbf{G} \in \mathbb{R}^7$
- decorrelate $\mathbf{G}$

Comparison with MFCC Computation



FFV versus Pitch Tracking, Conceptually



Related Work

- well-established pitch processing & modeling techniques
  - e.g., Shriberg & Stolcke, "Direct modeling of prosody: An overview of applications in automatic speech processing", *Speech Prosody 2004*.
- similar in purpose to $\Delta \log F_0$ **from cepstra**
  - K. Iwano et al, "Noise robust speech recognition using F_0 contour extracted by Hough transform", *ICSLP 2002*.
- algorithmically similar, for different task
  - P. Martin, "A fundamental frequency estimator by crosscorrelation of adjacent spectra", *Speech Prosody 2008*.

FFV for Speaker Change Prediction (ICASSP 2008)

- **CONTEXT:** Swedish Map Task (GIVER and FOLLOWER)
- **AUDIO:** 16kHz, close-talk, anechoic-chamber
- **GIVEN:** 500ms of speech at end-of-talkspurt, by GIVER
- **TASK:** predict whether GIVER will be next to speak

- **FINDINGS:**
  1. expected: GIVER appears to employ flat pitch to hold floor
  2. significantly outperm state-of-the-art pause-only system
  3. ML classifier outperforms a manually constructed state-of-the-art decision tree, which additionally uses pitch range information

FFV for Speaker Classification (ICASSP 2009)

- **CONTEXT:** read WSJ + some spontaneous utterances
- **AUDIO:** 16kHz, close-talk
- **GIVEN:** 1 minute interval of speech
- **TASK:** classify which of 100 people is the speaker

- **FINDINGS:**
  1. modeling speaker-dependent "intonation contour bias"
  2. model-space combination with MFCC features reduces error rates by > 40%rel

FFV for Floor Mechanism Classification (EUSIPCO 2009)

- **CONTEXT:** naturally occurring meetings, American English
- **AUDIO:** 16kHz, close-talk, lots of crosstalk
- **GIVEN:** manually determined 500ms at beg-/end-of-talkspurt
- **TASK:** classify whether a floor mechanism or other DA type

- **FINDINGS:**
  1. flat pitch at the beginning of floor mechanism talk
  2. slower speech at the end of floor mechanism talk

Basic but Relevant Questions

1. What are the effects of feature transformation?
2. What are the effects of feature regularization?
3. What are the effects of feature combination?
4. Can we reduce the computation time?
5. What are the effects of higher model complexity?

Will try to answer using classification accuracy and ROC area

- HMM log-likelihood-ratio classifier, 4 states, 1 Gaussian each
- task: floor DA type versus other DA type, balanced
- averaged over 4 individual context subtasks

Feature Transformation

- compare raw features with
  - global-Z-transformed features
  - global-PCA-rotated features

System		Acc	ROC
1	Baseline	64.8	71.5
2a	Z-Transform	65.7	73.8
2b	PCA Rotation	67.8	74.8

- feature transformation can be beneficial
- an optimal transform may be task- and audio- dependent

Feature Regularization

- replace response of 5 center filters with best parabolic fit

System		Acc	ROC
2b	New Baseline	67.8	74.8
3	Quadratic Fit	68.9	76.7

- the parabolic projection and application of the filterbank are linear operations
- tantamount to applying a modified filterbank
- an optimal filterbank may be task- and audio- dependent

Feature Combination

- concatenate with 5 “auxiliary” correlates of prosodic features:
  - log energy (loudness)
  - delta log energy (change in loudness)
  - normalized autocorrelation maximum (probability of voicing)
  - Mel-spectral flux (speaking rate)
  - log-Mel-spectral flux (speaking rate)

System		Acc	ROC
3	New Baseline	68.9	76.7
4a	Auxiliary Features	64.8	71.3
4b	Combination	69.6	77.6

- FFV features are complimentary to non-pitch prosodic features

Runtime Improvements

- most costly: extremity filter computation
- reduce support of extremity filters by a factor of >5

System		Acc	ROC
4b	New Baseline	69.6	77.6
5a	Exclusion of Extremity Filters	69.1	77.0
5b	Improvement of Extremity Filters	70.5	78.9

- runtime decreased by a factor of $\approx$5, to 0.27$\times$ at 2.5GHz
- unexpectedly: accuracy improvement also

GMM Model Complexity

- increase number of HMM states from 4
- increase number of Gaussians per HMM state from 1

System		Acc	ROC
5b	New Baseline	70.5	78.9
6	4 states, 2 Gaussians	71.7	80.3
7a	8 states, 1 Gaussian	71.3	79.4
7b	8 states, 2 Gaussians	73.0	81.5
7c	8 states, 3 Gaussians	73.3	82.4

- increasing model complexity helps
- appears to be a function of amount of data

Overview of Numerical Results

System		Acc	ROC
1	Baseline	**64.8**	**71.5**
2a	*Z*-Transform	65.7	73.8
2b	PCA Rotation	67.8	74.8
3	Quadratic Fit	68.9	76.7
4b	Combination with Auxiliary Features	69.6	77.6
5b	Improvement of Extremity Filters	70.5	78.9
6	4 states, 2 Gaussians	71.7	80.3
7b	8 states, 2 Gaussians	73.0	81.5
7c	8 states, 3 Gaussians	**73.3**	**82.4**
Relative reduction of error, %		**24.2**	**38.3**

A General-Purpose 32 ms Prosodic Vector for Hidden Markov Modeling

- **"32 ms Prosodic Vector"**
  - a short-time continuous representation of variation in F_0
  - can be combined with standard short-time features
- **"General-Purpose"**
  - applicability across multiple tasks
- **"for Hidden Markov Modeling"**
  - feature transformation and regularization are helpful
  - performance improves as model complexity increases
  - real-time factor is 0.27
- **normative implementation**
  - `www.cs.cmu.edu/~kornel/software.html`