View this talk on YouTube: https://youtu.be/sI_8EA0_ha8

Training neural network acoustic models on (multichannel) waveforms

Ron Weiss

in SANE 2015

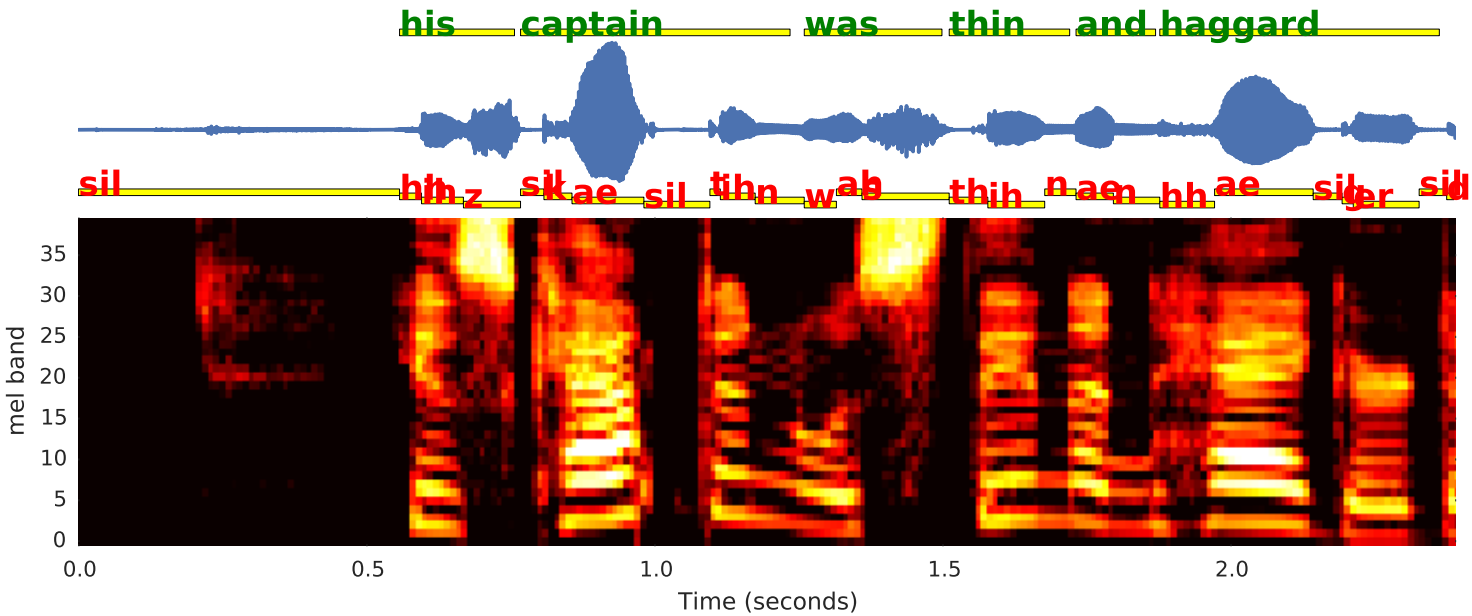
2015-10-22

Joint work with Tara Sainath, Kevin Wilson, Andrew Senior, Arun Narayanan, Michiel Bacchiani, Oriol Vinyals, Yedid Hoshen

Outline

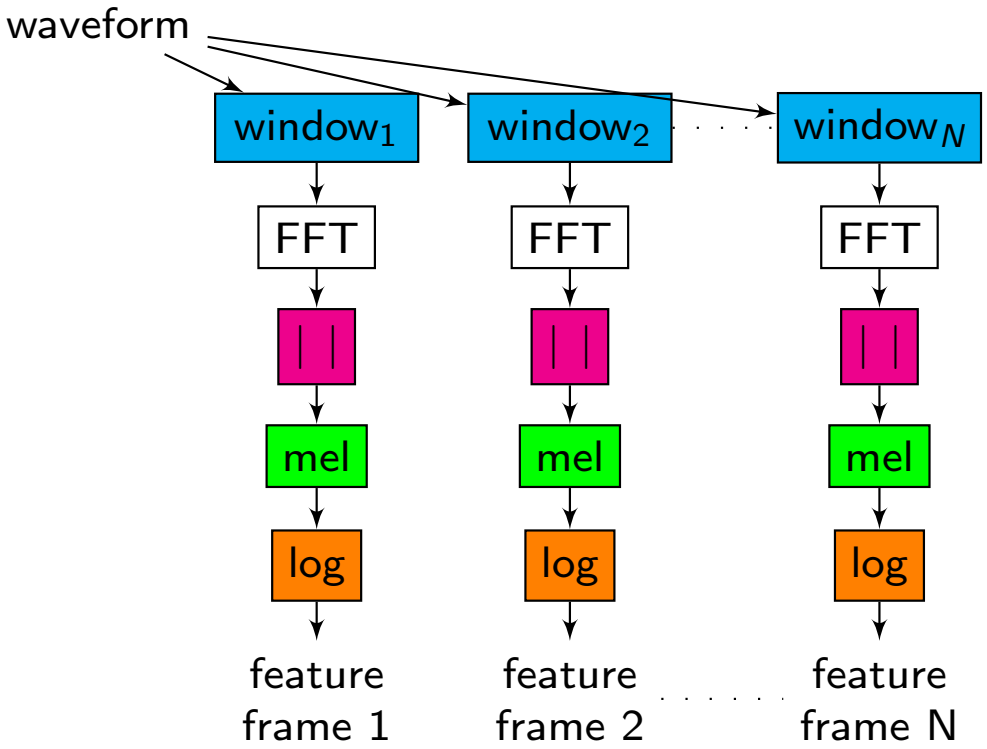
2. Sainath, T. N., Weiss, R. J., Senior, A., Wilson, K. W., and Vinyals, O. (2015b). Learning the speech front-end with raw waveform CLDNNs. In *Proc. Interspeech*
4. Sainath, T. N., Weiss, R. J., Wilson, K. W., Narayanan, A., Bacchiani, M., and Senior, A. (2015c). Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms. In *Proc. ASRU*. to appear

Acoustic modeling in 2015



- Classify each 10ms audio frame into context-dependent phoneme state
- Log-mel filterbank features passed into a neural network
- Modern vision models are trained directly from the pixels, can we train an acoustic model directly from the samples?

Frequency domain filterbank: log-mel



localization in time

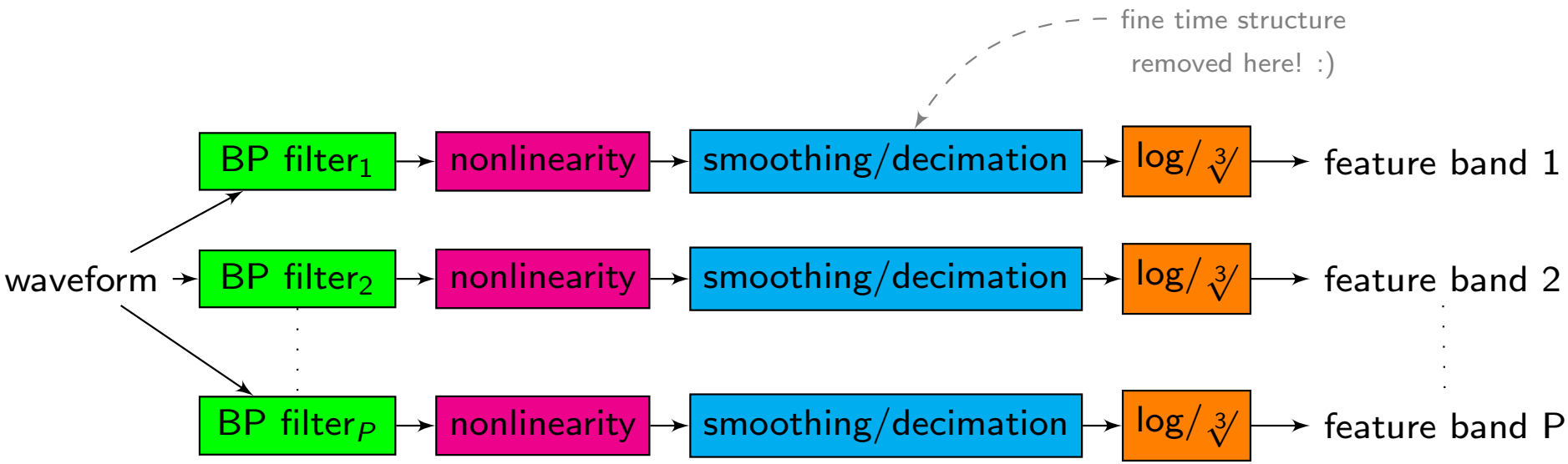
pointwise nonlinearity

bandpass filtering

dynamic range compression

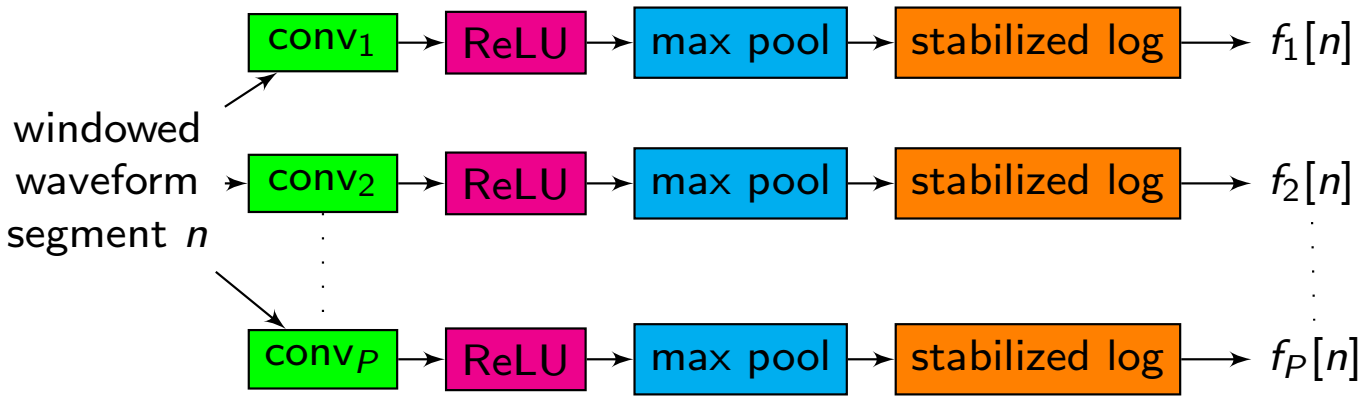
Bandpass filtering implemented using FFT and mel warping

Time-domain filterbank



- Swap order of filtering and decimation, but basically the same thing
- Cochleagrams, gammatone features for ASR (Schluter et al., 2007)

Time-domain filterbank as a neural net layer

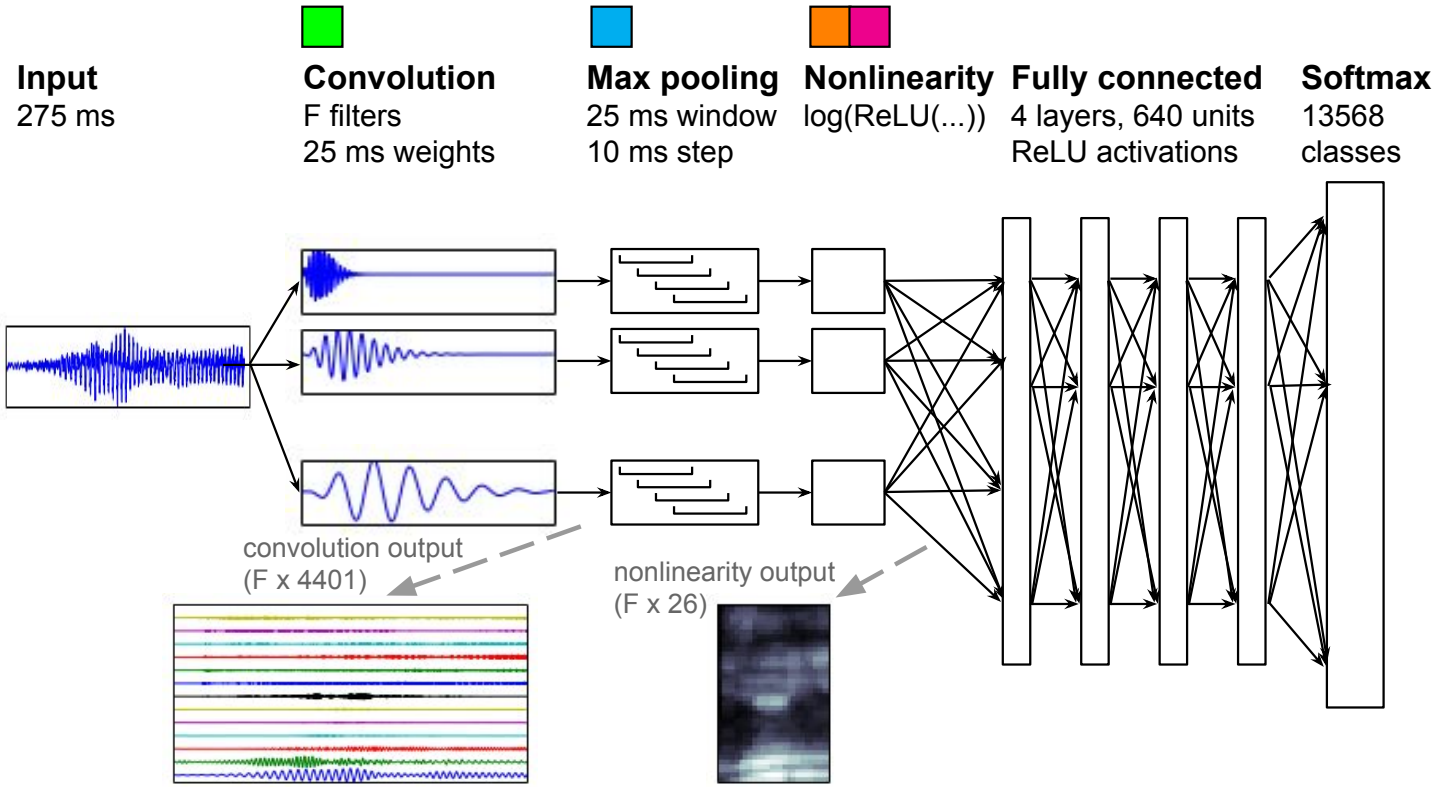


- These are common neural network operations
  - (FIR) filter $\rightarrow$ convolution
  - nonlinearity $\rightarrow$ rectified linear (ReLU) activation
  - smoothing/decimation $\rightarrow$ pooling
- Window waveform into short ($<$ 300ms) overlapping segments
- Pass each segment into FIR filterbank to generate feature frame

Previous work: Representation learning from waveforms

- Jaitly and Hinton (2011)
  - unsupervised representation learning using a time-convolutional RBM
  - supervised DNN training on learned features for phone recognition
- Tüske et al. (2014), Bhargava and Rose (2015)
  - supervised training, fully connected DNN
  - learns similar filter shapes at different shifts
- Palaz et al. (2013, 2015b,a), Hoshen et al. (2015), Golik et al. (2015)
  - supervised training, convolution to share parameters across time shifts
- No improvement over log-mel baseline on large vocabulary task in above work

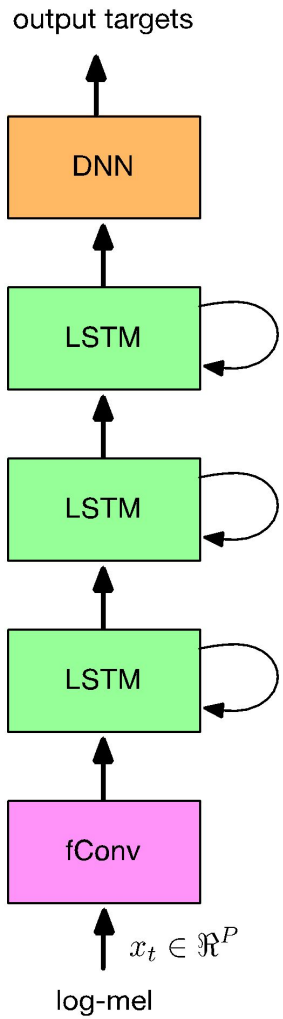
Deep waveform DNN (Hoshen et al., 2015)



- Choose parameters to match log-mel DNN
  - 40 filters, 25ms impulse response, 10 ms hop
  - stack 26 frames of context using strided pooling: 40x26 "brainogram"
- Adding stabilized log compression gave 3-5% relative WER decrease
- Overall 5-6% relative WER increase compared to log-mel DNN

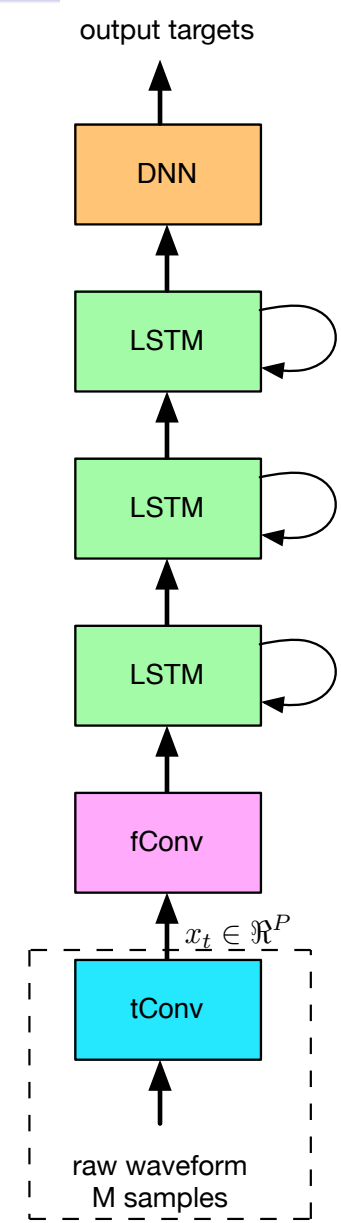
CLDNN (Sainath et al., 2015a)

- Combine all the neural net tricks:
  CLDNN = Convolution + LSTM + DNN
  - Frequency convolution gives some pitch/vocal tract length invariance
  - LSTM layers model long term temporal structure
  - DNN learns linearly separable function of LSTM state
- $4 - 6\%$ improvement over LSTM baseline
- No need for extra frames of context in input:
  memory in LSTM can remember previous inputs



Waveform CLDNN (Sainath et al., 2015b)

- Time convolution (tConv) produces a 40dim frame
  - 35ms window ($M = 561$ samples), hopped by 10ms
- CLDNN similar to (Sainath et al., 2015a)
  - Frequency convolution (fConv) layer:
    - 8x1 filter, 256 outputs, pool by 3 without overlap
    - 8x256 output fed into linear dim reduction layer
  - 3 LSTM layers:
    - 832 cells/layer with 512 dim projection layer
  - DNN layer:
    - 1024 nodes, ReLU activations
    - linear dim reduction layer with 512 outputs
- Total of 19M parameters, 16K in tConv
- All trained jointly with tConv filterbank



Experiments

- US English Voice Search task,
  1. *Clean* dataset: 3M utterances ($\sim 2k$ hours) train, 30k (~ 20 hours) test
  2. *MTR20* multicondition dataset:
     simulated noise and reverberation
     - SNR between 5-25dB (average $\sim$ 20dB)
     - RT_{60} between 0-400ms (average $\sim$ 160ms)
     - Target to mic distance between 0-2m (average $\sim$ 0.75m)
- 13522 context-dependent state outputs
- Asynchronous SGD training, optimizing a cross-entropy loss

Compared to log-mel (Sainath et al., 2015b)

Train/test set	Feature	WER
Clean	log-mel	14.0
	waveform	**13.7**
MTR20	log-mel	16.2
	waveform	16.2
	waveform+log-mel	**15.7**

- Matches performance of log-mel baseline in clean and moderate noise
- 3% relative improvement by stacking log-mel features and tConv output

How important are LSTM layers? (Sainath et al., 2015b)

Architecture	MTR20 WER log-mel	MTR20 WER waveform
D6	22.3	23.2
F1L1D1	17.3	17.8
F1L2D1	16.6	16.6
F1L3D1	**16.2**	**16.2**

- Fully connected DNN: waveform 4% worse than log-mel
- Log-mel outperforms waveform with one or zero LSTM layers
- Time convolution layer gives short term shift invariance, but seems to need recurrence to model longer time scales

Bring on the noise (Sainath et al., 2015c)

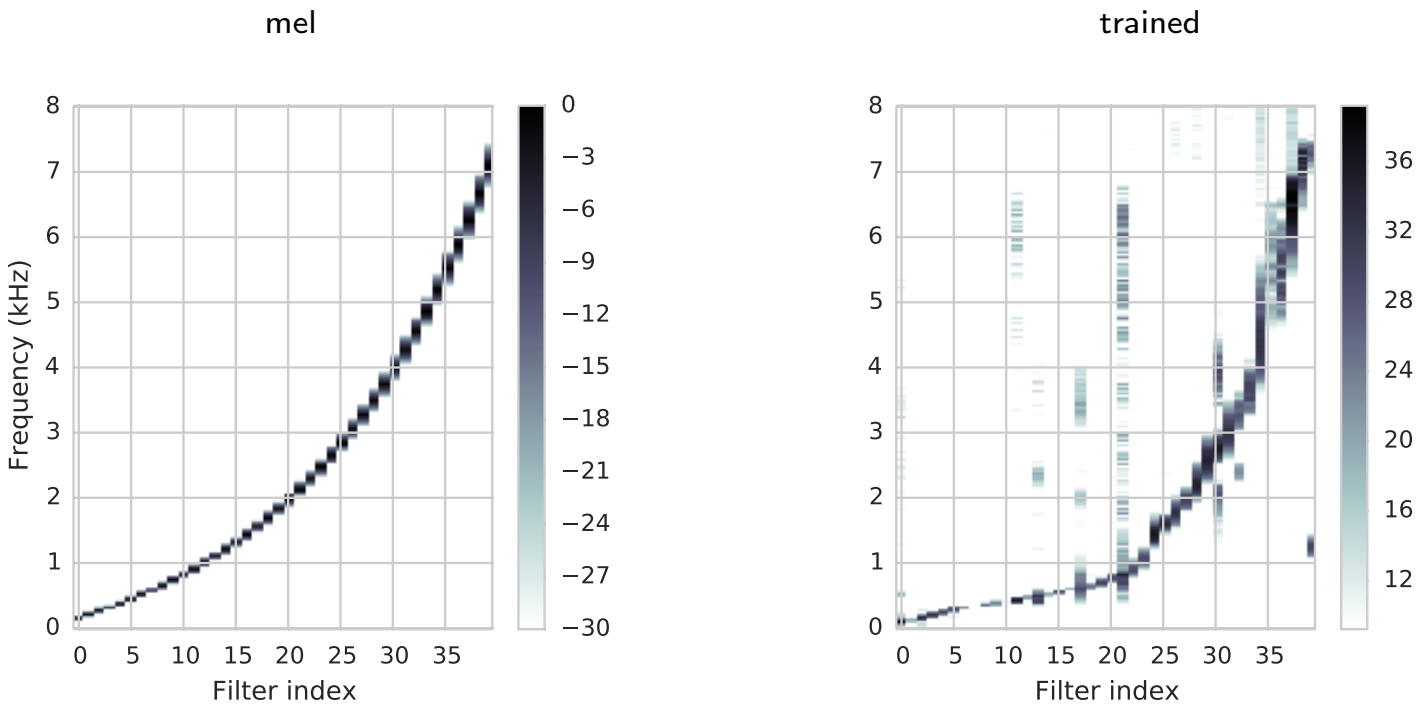
MTR12: noisier version of MTR20

- 12dB average SNR, 600ms average RT_{60}, more farfield

Num filters	log-mel	waveform
40	25.2	**24.7**
84	25.0	**23.7**
128	24.4	**23.5**

- Waveform consistently outperforms log-mel in high noise
- Larger improvements with more filters

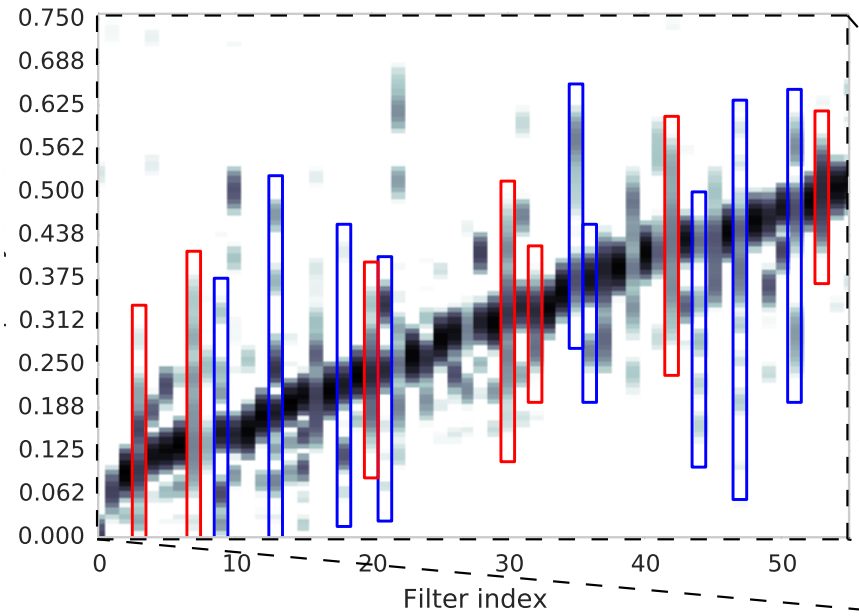
Filterbank magnitude responses

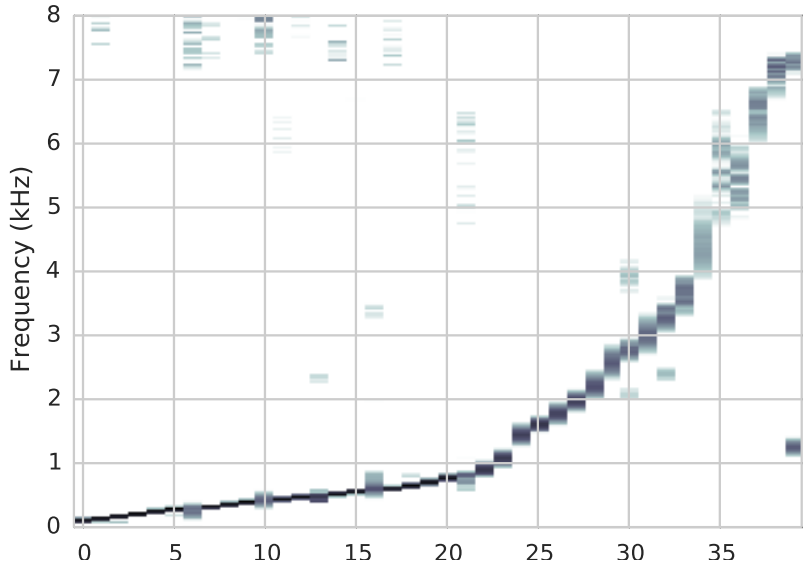


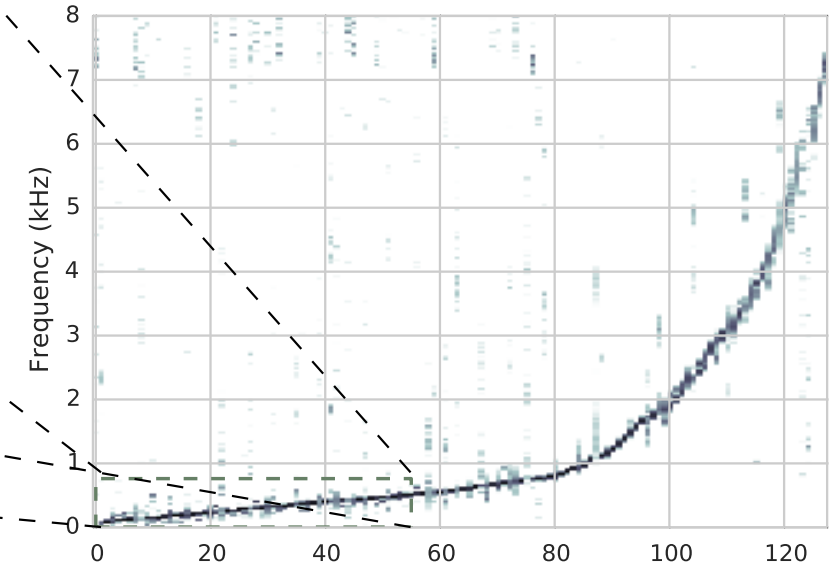
- Sort filters by index of frequency band with peak magnitude
- Looks mostly like an auditory filterbank
  - mostly bandpass filters, bandwidth increases with center frequency
- Consistently higher resolution in low frequencies:
  - 20 filters below 1kHz vs ∼10 in mel
  - somewhat consistent with an ERB auditory frequency scale

What happens when we add more filters?

- $>$ 80 filters below 1kHz: *overcomplete basis*
- Not all bandpass anymore
  - harmonic stacks
  - wider bandwidths





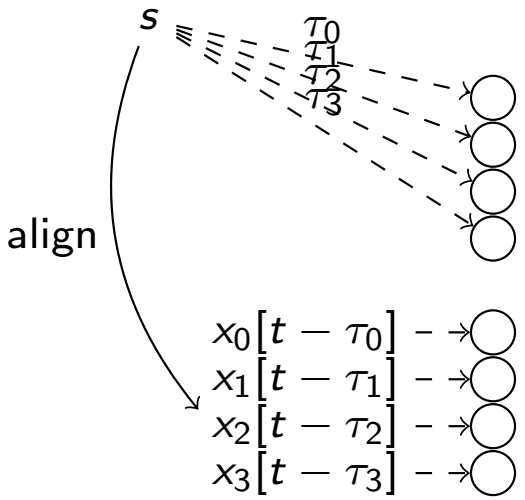


What if we had a microphone array...

- Build a noise robust multichannel ASR system by cascading:
  1. speech enhancement to reduce noise
     e.g. localization + beamforming + nonlinear postfiltering
  2. acoustic model, possibly trained on the output of 1
- Perform multichannel enhancement and acoustic modeling *jointly*?
  - Seltzer et al. (2004) explored this idea using a GMM acoustic model
  - we're going to use neural nets

Filter-and-sum beamforming

$$y[t] = \sum_{c=0}^{C-1} h_c[t] * x_c[t - \tau_c]$$

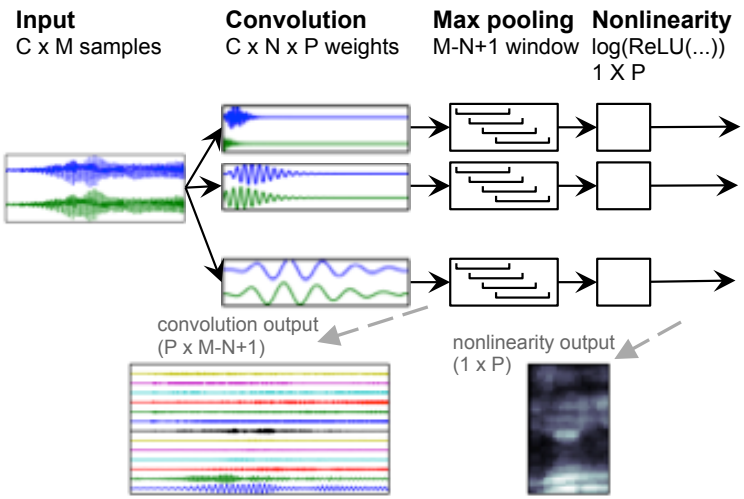


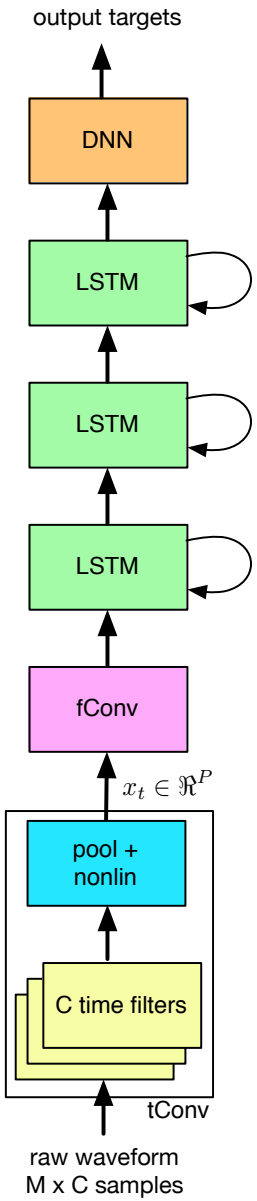
- Typical to have separate localization model estimate τ_c, and a beamformer estimate filter weights
- Use P filters to capture many *fixed* steering delays

$$y^p[t] = \sum_{c=0}^{C-1} h_c^p[t] * x_c[t]$$

- Just another convolution across a *multichannel* waveform

Multichannel waveform CLDNN (Sainath et al., 2015c)





- Multichannel tConv layer
  - bank of filter-and-sum beamformers, but without explicit localization and alignment
  - does both *spatial* and *spectral* filtering
- Feeds into same CLDNN as in single channel case

Experiments

- MTR12 dataset, but *simulating* an 8 channel linear mic array
- Look at different microphone subsets
  - 1 channel: mic 1
  - 2 channel: mics 1,8: 14cm spacing
  - 4 channel: mics 1,3,6,8: 4cm-6cm-4cm spacing
  - 8 channel: mics 1-8: 2cm spacing
- 100 different room configurations
- Noise and target speaker location randomly selected for each utterance
- Main test set with same conditions as training

Compared to log mel (Sainath et al., 2015c)

Input	Num filters	1ch	2ch	4ch	8ch
log-mel	128	24.4	22.0	21.7	22.0
waveform	128	**23.5**	21.8	21.3	21.1
waveform	256	-	**21.7**	**20.8**	**20.6**

- Log-mel improves with additional channels (stack features from each channel) (Swietojanski et al., 2013) but not as much as waveform
  - fine time structure discarded with the phase
- Waveform improvements saturate at 128 filters with 2 channels
- Continue to see improvements with 256 filters with 4 and 8 channels
  - can learn more complex spatial responses with more microphones, allowing net to make good use of extra capacity in filterbank layer

How many LSTM layers does it take?

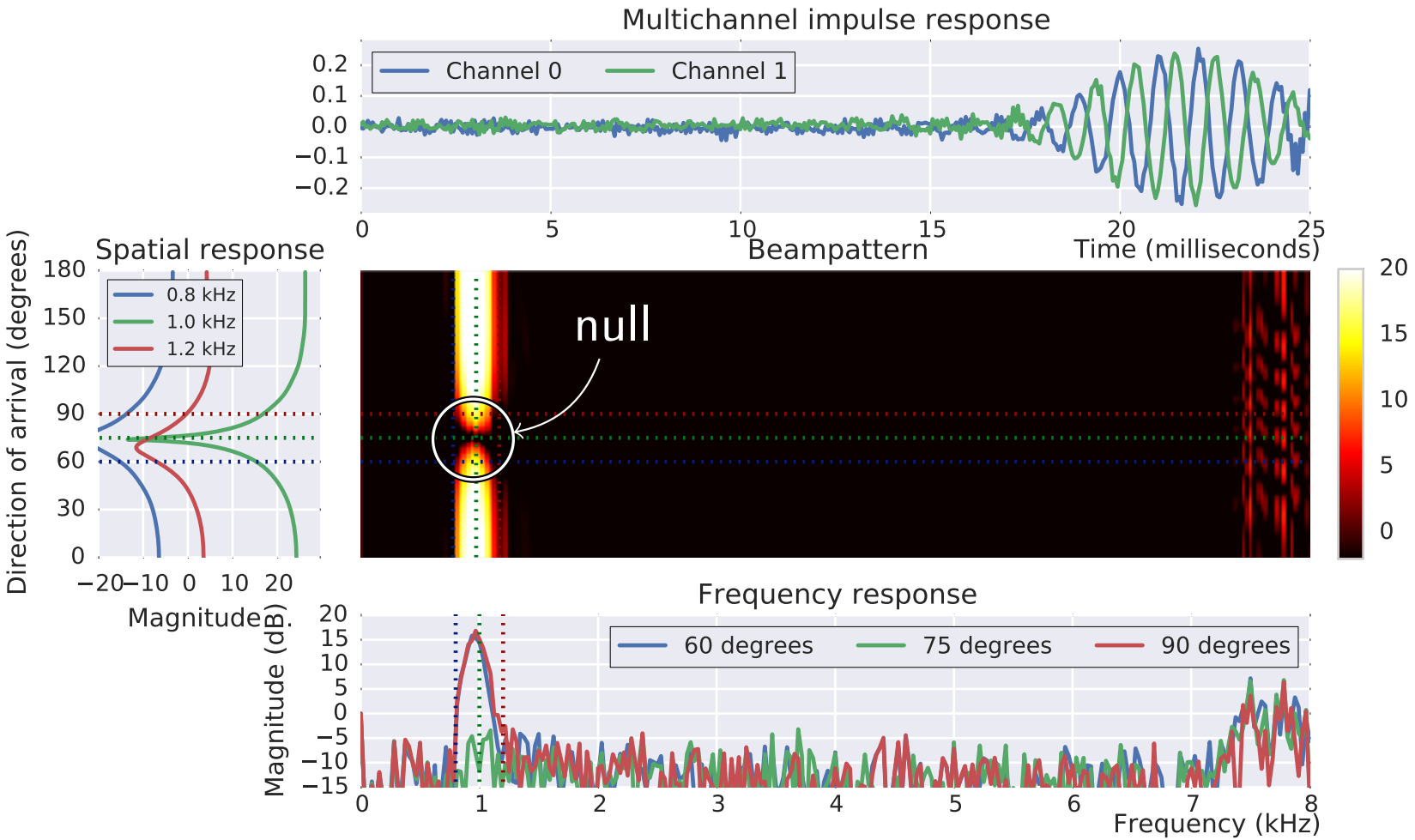
Input	Num filters	Num LSTM layers	WER
waveform, 2ch	128	1	25.8
waveform, 2ch	128	2	23.9
waveform, 2ch	128	3	21.8
waveform, 2ch	128	4	**21.5**

- As in 1 channel case, modeling temporal context with LSTM layers is key to getting good performance
- Starts to saturate at 3 LSTM layers

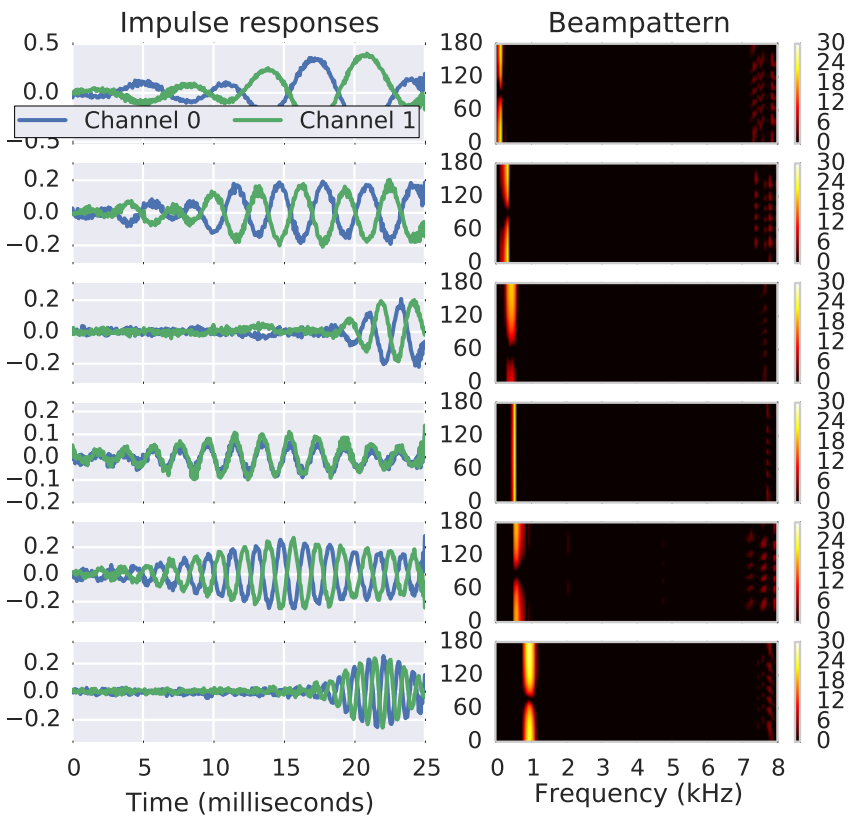
What's a Beampattern?!

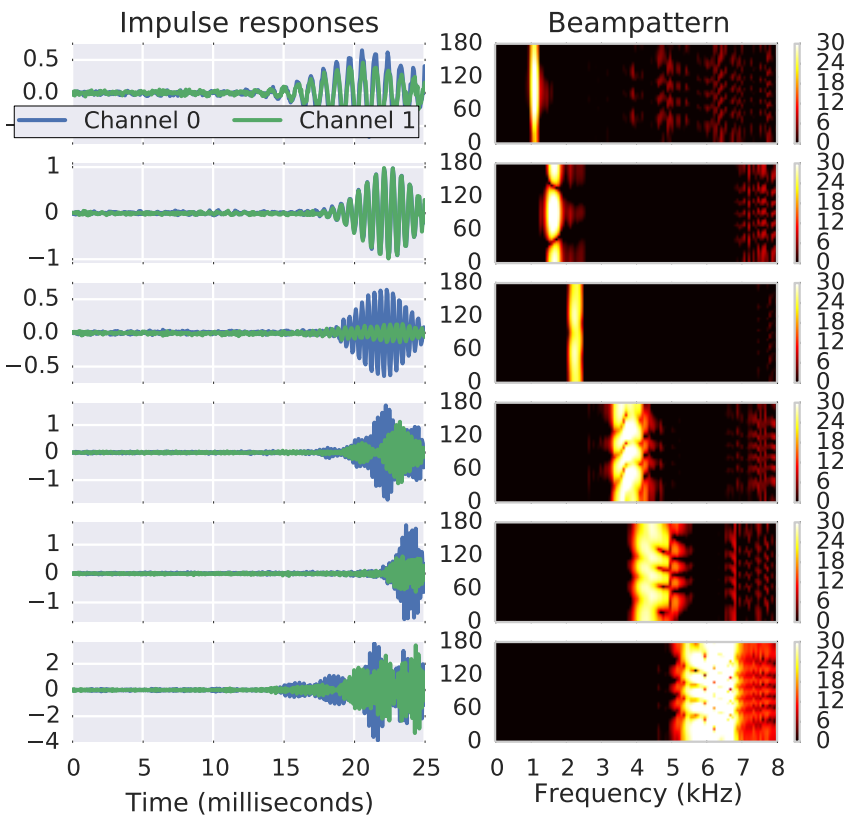
Magnitude response as a function of direction of arrival to microphone array

- pass "multimic impulse" with different delays into filter, measure resp.



What is this thing learning? Example filters

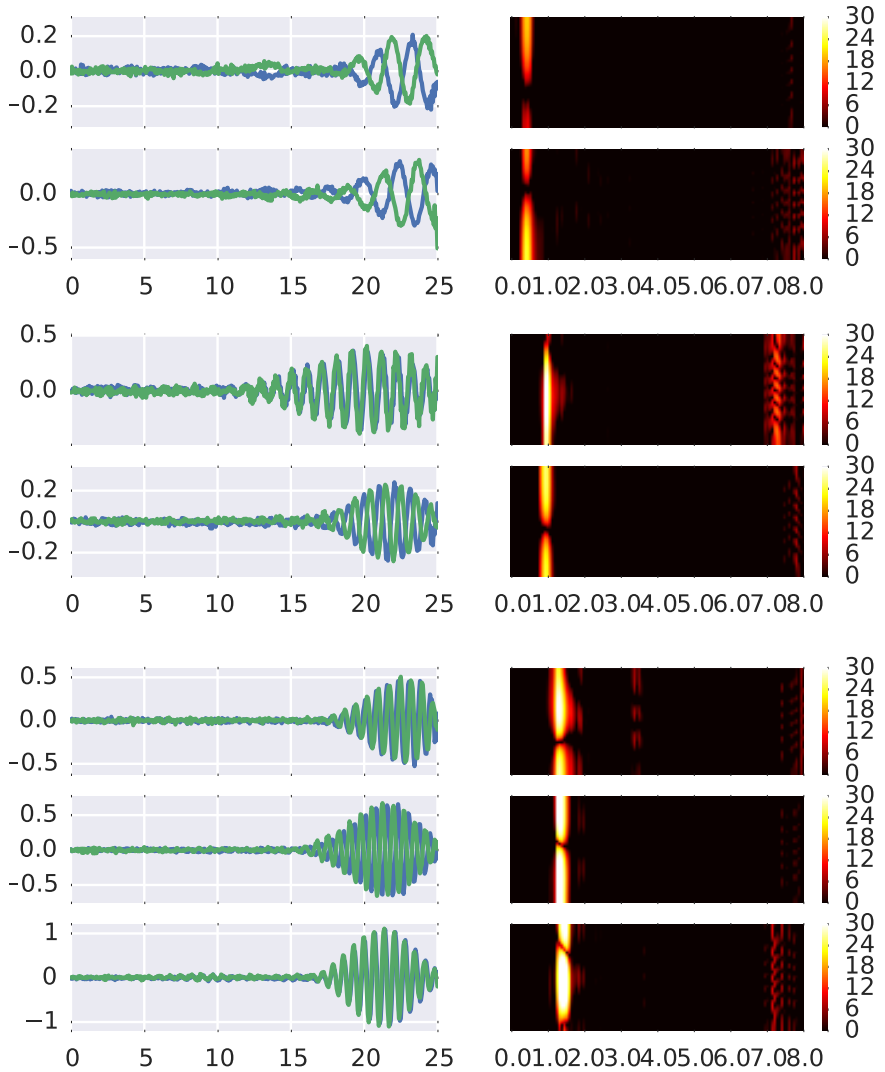




- Similar coefficients across channels but shifted, similar to steering delay
- Most filters have bandpass freq. response, similar scale to 1ch
- $\sim$ 80% of the filters have a significant spatial response

Even more example filters

- Several filters with the same center frequency, different null directions
- Enables upstream layers to differentiate between energy coming from different directions in narrow bands



Compared to traditional beamforming (Sainath et al., 2015c)

System	2ch	4ch	8ch
oracle D+S	22.8	22.5	22.4
waveform	**21.8**	**21.3**	**21.1**

- Delay-and-sum (D+S) baseline using oracle time difference of arrival, passed into 1ch waveform model
- Despite lack of explicit localization waveform outperforms D+S
  - upper layers learn invariance to direction of arrival?

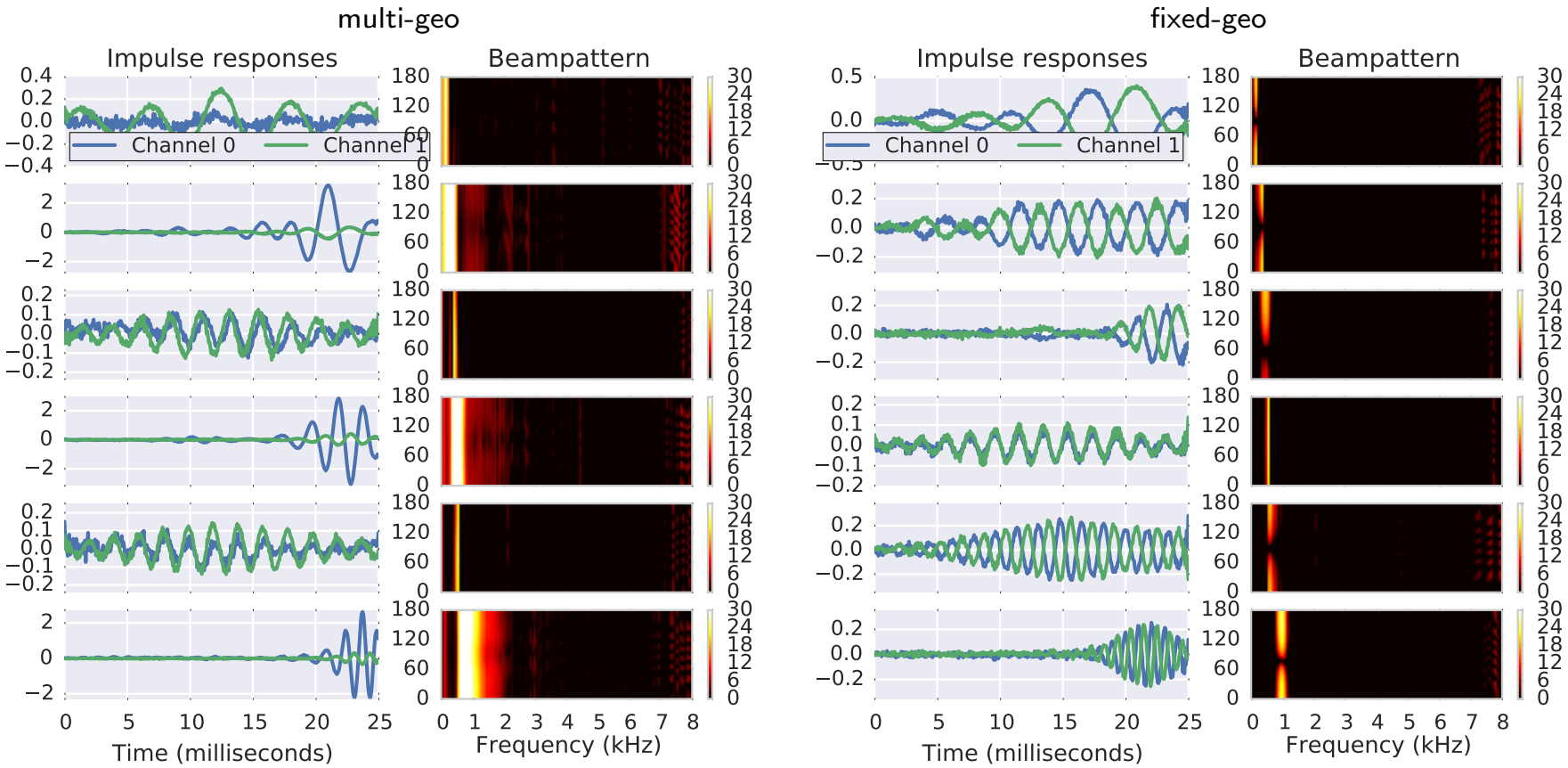
Mismatched array geometry (Sainath et al., 2015c)

System	Spacing 14cm	10cm	6cm	2cm	0cm[1]
oracle D+S 2ch	22.8	23.2	23.3	23.7	23.5
waveform 2ch, 14cm	**21.8**	22.2	23.3	30.7	33.9
waveform 2ch, multi-geo	21.9	**21.7**	**21.9**	**21.8**	**23.1**

- Oracle D+S more robust to mismatches in microphone spacing
- Degraded performance if mic array spacing differs widely from training
- "Multi-geometry" training set by sampling 2 channels with replacement for each utterance in the original 8 channel set
  - net trained on this data becomes **invariant to microphone spacing**
  - also **robust** to decoding a single channel?!

[1] repeat signal from mic 1

Multigeometrained filters



- Still get bandpass filters, but *without strong spatial responses*
  - only 30% of the filters have a null
  - several filters primarily respond to only one channel
- Upper layers of the network somehow learn to model directionality?

Mismatched test (Sainath et al., 2015c)

System	Simulated (14cm)	Rerecorded (28cm)
waveform, 1ch	19.3	23.8
waveform, 2ch, 14cm	18.2	23.7
oracle D+S, 2ch	19.2	23.3
waveform, 2ch, multi-geo	**17.8**	**21.1**

*after sequence training

- Slightly more realistic "Rerecorded" test set:
  - replay sources from eval set through speakers in a living room
  - record using an 8-channel linear microphone array with 4cm spacing
  - artificially mixed using same SNR distribution as MTR12set
- Multigeometraining still works

Conclusion

From feature engineering to... deep net architecture engineering:

- Supervised training to learn filter coefficients, optimized jointly with target objective
- Waveform CLDNN matches log-mel on clean, outperforms it on noisy
- Larger performance improvement with multichannel input
- Secret sauce: LSTM layers
- Multicondition training/data augmentation work really well: clean *and* noisy, various mic array spacings

References I

Bhargava, M. and Rose, R. (2015). Architectures for deep neural network based acoustic models defined over windowed speech waveforms. In *Proc. Interspeech*.

Golik, P., Tüske, Z., Schlüter, R., and Ney, H. (2015). Convolutional neural networks for acoustic modeling of raw time signal in LVCSR. In *Proc. Interspeech*.

Hoshen, Y., Weiss, R. J., and Wilson, K. W. (2015). Speech Acoustic Modeling from Raw Multichannel Waveforms. In *Proc. ICASSP*.

Jaitly, N. and Hinton, G. (2011). Learning a better representation of speech soundwaves using restricted Boltzmann machines. In *Proc. ICASSP*.

Palaz, D., Collobert, R., and Magimai.-Doss, M. (2013). Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks. In *Proc. Interspeech*.

Palaz, D., Magimai.-Doss, M., and Collobert, R. (2015a). Analysis of CNN-based speech recognition system using raw speech as input. In *Proc. Interspeech*.

Palaz, D., Magimai.-Doss, M., and Collobert, R. (2015b). Convolutional neural networks-based continuous speech recognition using raw speech signal. Technical report.

Sainath, T. N., Vinyals, O., Senior, A., and Sak, H. (2015a). Convolutional, long short-term memory, fully connected deep neural networks. In *Proc. ICASSP*.

Sainath, T. N., Weiss, R. J., Senior, A., Wilson, K. W., and Vinyals, O. (2015b). Learning the speech front-end with raw waveform CLDNNs. In *Proc. Interspeech*.

Sainath, T. N., Weiss, R. J., Wilson, K. W., Narayanan, A., Bacchiani, M., and Senior, A. (2015c). Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms. In *Proc. ASRU*. to appear.

Schluter, R., Bezrukov, L., Wagner, H., and Ney, H. (2007). Gammatone features and feature combination for large vocabulary speech recognition. In *Proc. ICASSP*.

Seltzer, M. L., Raj, B., and Stern, R. M. (2004). Likelihood-maximizing beamforming for robust hands-free speech recognition. *IEEE Transactions on Speech and Audio Processing*, 12(5):489–498.

Swietojanski, P., Ghoshal, A., and Renals, S. (2013). Hybrid acoustic models for distant and multichannel large vocabulary speech recognition. In *Proc. ASRU*, pages 285–290.

Tüske, Z., Golik, P., Schlüter, R., and Ney, H. (2014). Acoustic modeling with deep neural networks using raw time signal for LVCSR. In *Proc. Interspeech*.

5 Extra slides

Even more multicondition training

Input	Train set	Test set Clean	Test set MTR20	Test set MTR12
log-mel	MTR20	**10.9**	12.6	25.8
log-mel	MTR12	13.4	14.6	19.6
log-mel	MTR20+12	11.1	**12.3**	19.6
waveform	MTR12	13.7	14.5	18.6
waveform	MTR20+12	**11.0**	12.6	**18.4**

*after sequence training

- Training on very noisy data hurts performance in clean
- CLDNNs have a lot of capacity:
  Training on both recovers clean performance, still works well on noisy

Why does this work? tConv / pooling (Sainath et al., 2015b)

Input window size	Pooling	Initialization	MTR20 WER
25ms	none	random	19.9
35ms	max	random	16.4
35ms	max	gammatone fixed	16.4
35ms	max	gammatone	**16.2**
35ms	l_2	gammatone	16.4
35ms	average	gammatone	16.8

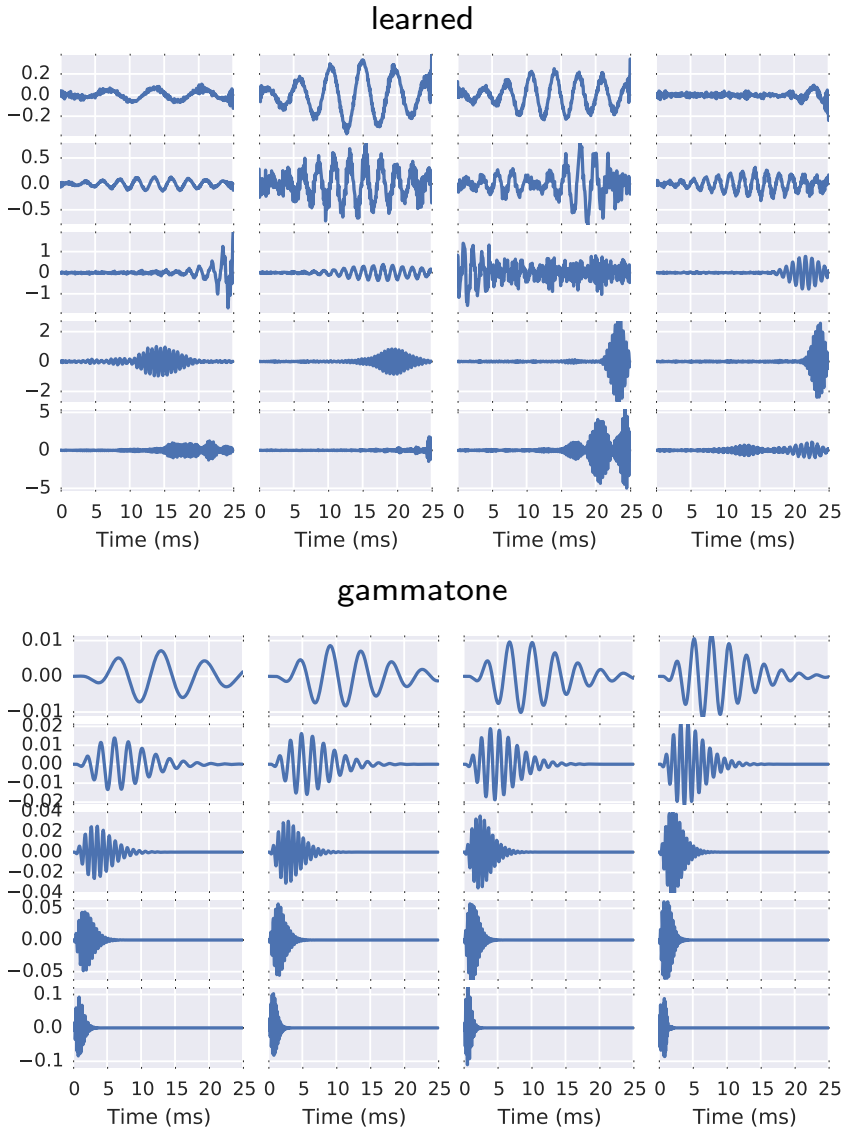
- Pooling gives shift invariance over short (35 - 25 = 10ms) time scale
- Poor performance without pooling - fixed phase
- Best results with (ERB) gammatone initialization and max pooling
  - because of filter ordering assumed by fConv?
  - max preserves transients smoothed out by other pooling functions?
- Not training tConv layer is slightly worse

How important is frequency convolution? (Sainath et al., 2015b)

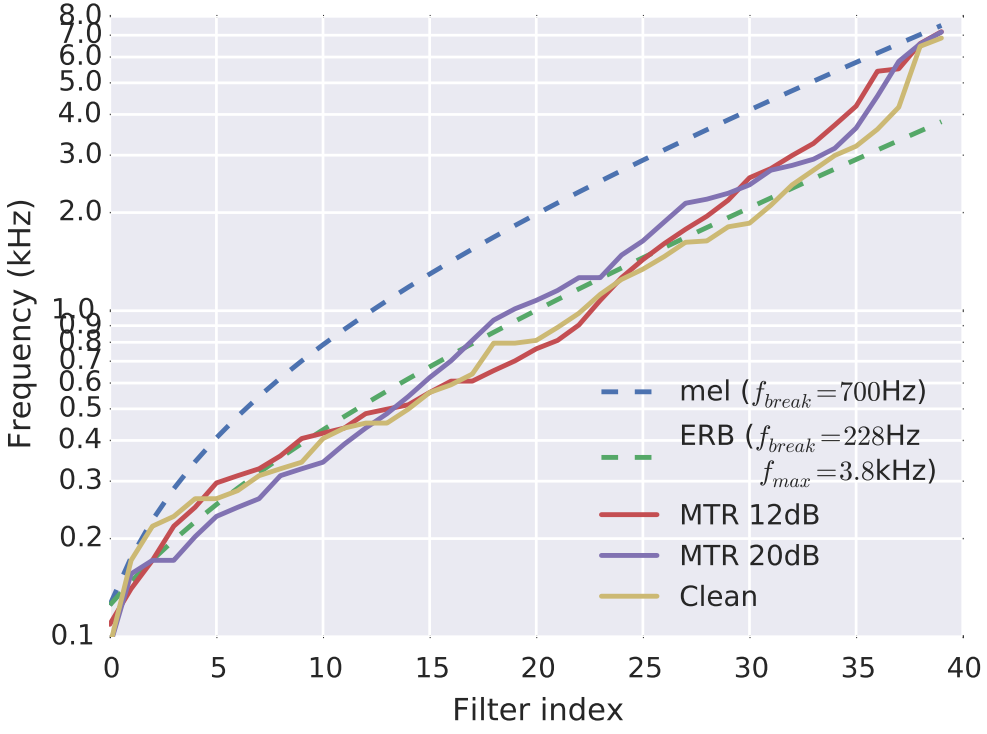
Input	Architecture	MTR20 WER
log-mel	F1L3D1	**16.2**
waveform	F1L3D1	**16.2**
log-mel	L3D1	16.5
waveform	L3D1	16.5
waveform	L3D1, rand init	16.5

- Analyze results for different FxLyDz architectures
- Log-mel and waveform match performance if we remove fConv layer
- No difference in performance when randomly initializing tConv layer
  - fConv layer requires ordering of features coming out of tConv layer

Filterbank impulse responses

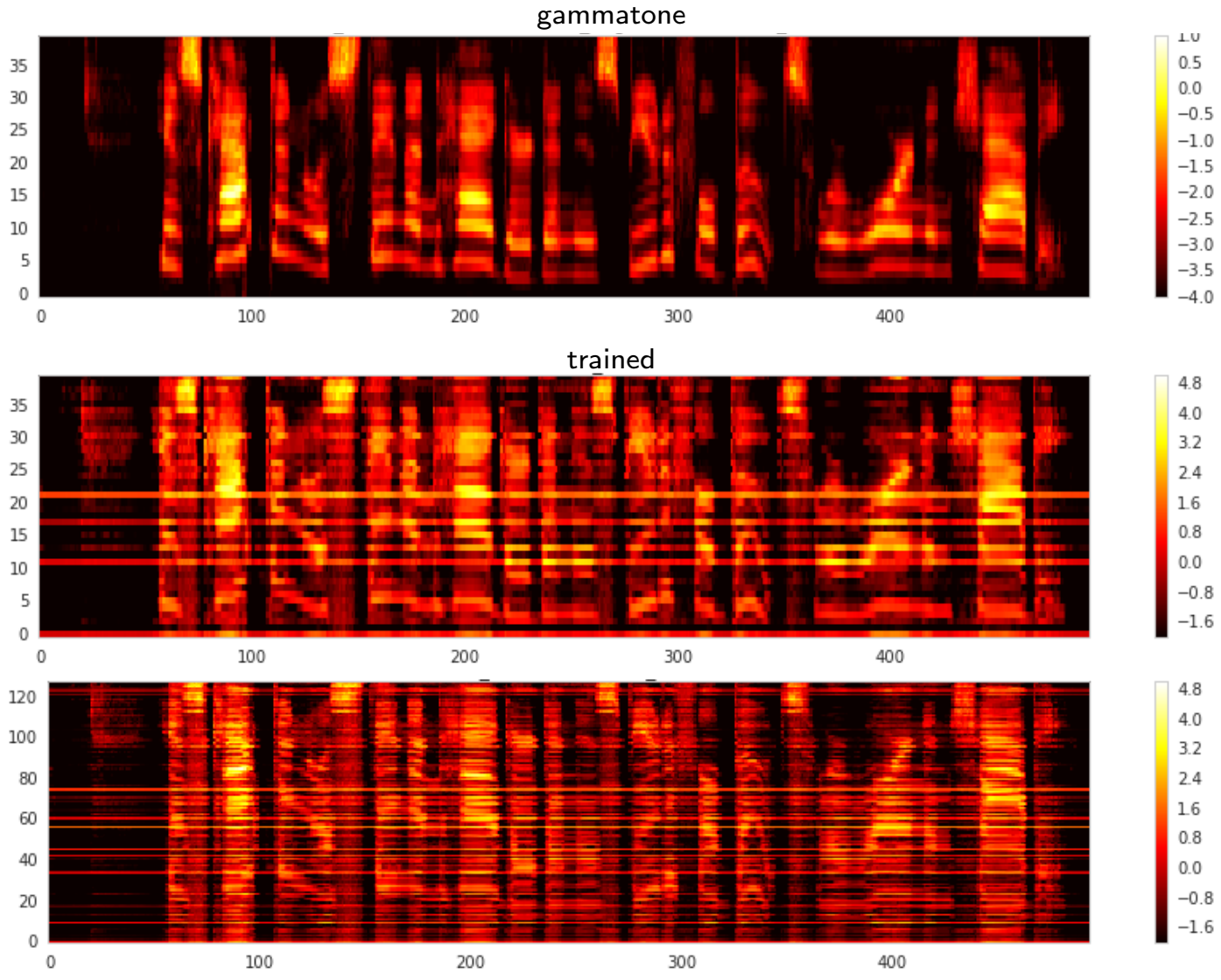


Does it correspond to an auditory frequency scale?

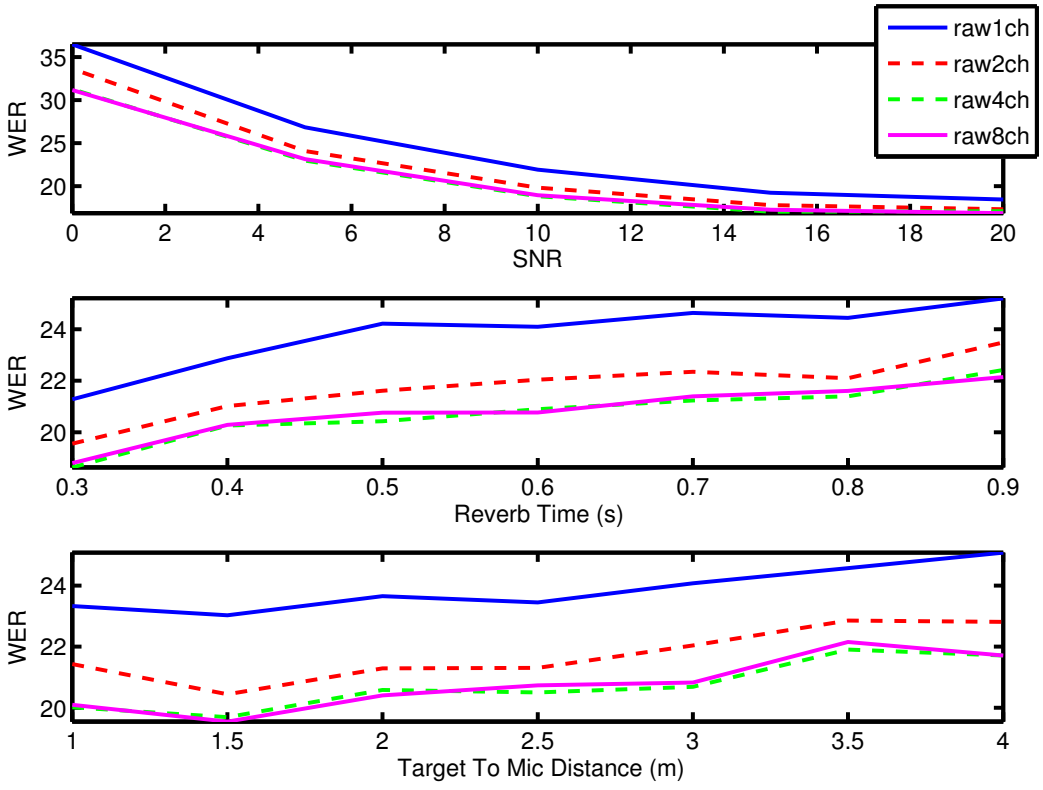


- Dick Lyon on mel spectrograms:
  "their amplitude scale is too logarithmic, and their frequency scale not logarithmic enough"
- Deep learning agrees: scale consistent with ERB spanning 3.8kHz
- Except it adds $\sim$ 5 filters above 4kHz

Single channel "brainograms"



Multichannel WER breakdown (Sainath et al., 2015c)



- Larger improvements at lowest SNRs
- Consistent improvements across range of reverb times and target distances

Compared to traditional beamforming (Sainath et al., 2015c)

- Compare waveform model to two baselines
  1. delay-and-sum (D+S) using oracle time difference of arrival (TDOA), passed into 1ch waveform model
  2. time-aligned multichannel (TAM) using oracle TDOA, passed into multichannel waveform model

System	2ch	4ch	8ch
oracle D+S	22.8	22.5	22.4
oracle TAM	**21.7**	**21.3**	21.3
waveform	21.8	**21.3**	**21.1**

- Despite lack of explicit localization waveform does better than D+S, matches TAM
  - upper layers learn invariance to direction of arrival?
- TAM learns filters similar to "uncompensated" waveform