AUDIO FEATURES & MACHINE LEARNING

E.M. Bakker

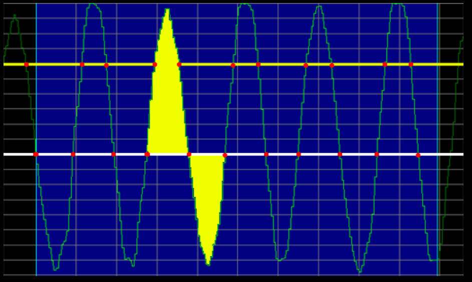
API2022

FEATURES FOR SPEECH RECOGNITION AND AUDIO INDEXING

- Parametric Representations
  - Short Time Energy
  - Zero Crossing Rates
  - Level Crossing Rates
  - Short Time Spectral Envelope
- Spectral Analysis
  - Filter Design
  - Filter Bank Spectral Analysis Model
  - Linear Predictive Coding (LPC)
  - MFCCs

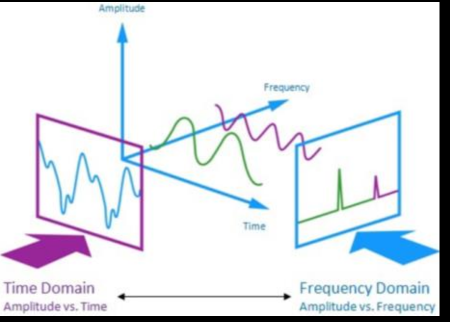
FEATURES FOR SPEECH RECOGNITION AND AUDIO INDEXING

- Parametric Representations
  - Short Time Energy
  - Zero Crossing Rates
  - Level Crossing Rates

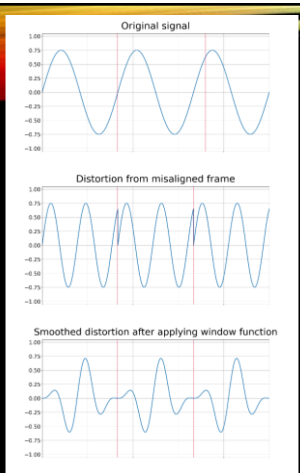


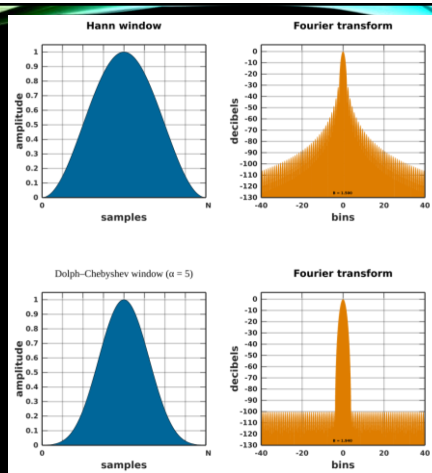
Example: Speech of length 0.01 sec.

FEATURES FOR SPEECH RECOGNITION AND AUDIO INDEXING



- Spectral Analysis
  - Fourier Transform
  - Filter Design
  - Filter Bank Spectral Analysis Model
  - Linear Predictive Coding (LPC)
  - MFCCs

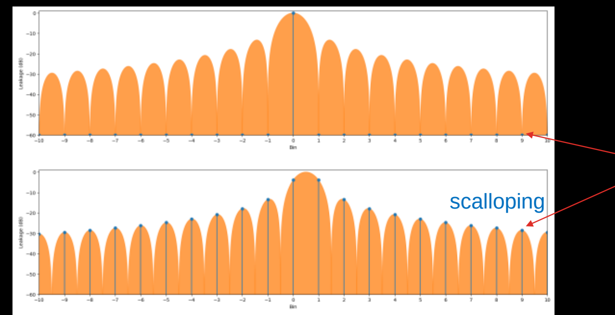




By L. de Jonckheere

- Spectral Analysis
  - Fourier Transform
  - Rectangular window => high resolution, low dynamic range (not good at distinguishing components of different amplitudes)
  - Hann or Hamming window => moderate

FEATURES FOR SPEECH RECOGNITION AND AUDIO INDEXING



By L. de Jonckheere

- Spectral Analysis
  - Fourier Transform
  - Frame of samples => frequency bins
  - Each bin corresponds to one frequency

  => Spectral leakage

SHORT TIME FOURIER TRANSFORM SHORT HAMMING WINDOW: 50 SAMPLES (=5MSEC)

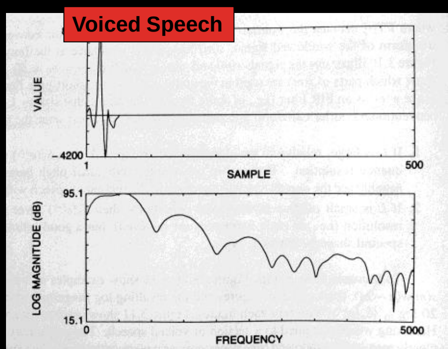


Figure 3.12 Short-time Fourier transform using a short (50 points or 5 msec) Hamming window on a section of voiced speech.

From: Rabiner et al.

SHORT TIME FOURIER TRANSFORM LONG HAMMING WINDOW: 500 SAMPLES (=50MSEC)

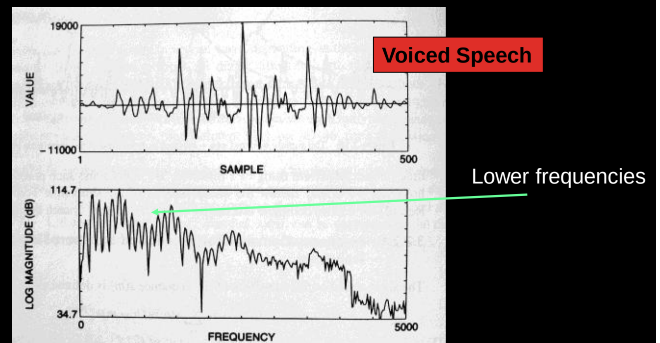


Figure 3.11 Short-time Fourier transform using a long (500 points or 50 msec) Hamming window on a section of voiced speech.

From: Rabiner et al.

SHORT TIME FOURIER TRANSFORM SHORT HAMMING WINDOW: 50 SAMPLES (=5MSEC)

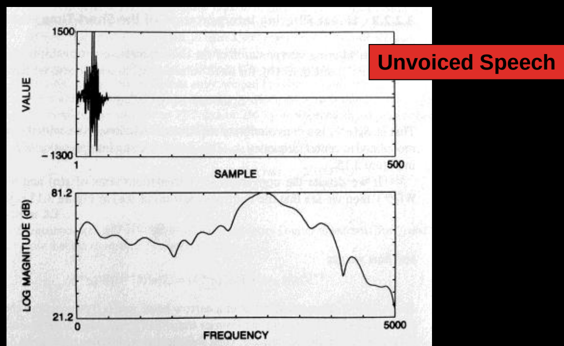


Figure 3.14 Short-time Fourier transform using a short (50 points or 5 msec) Hamming window on a section of unvoiced speech.

From: Rabiner et al.

SHORT TIME FOURIER TRANSFORM LONG HAMMING WINDOW: 500 SAMPLES (=50MSEC)

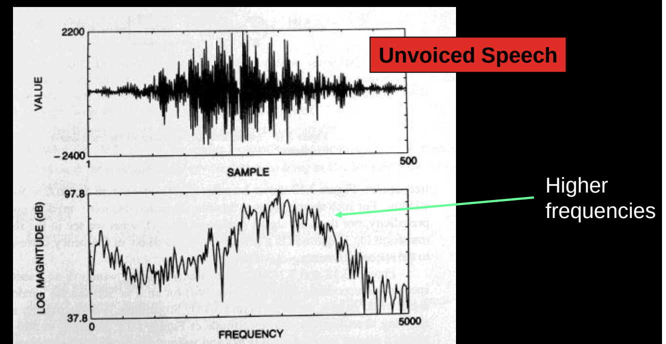
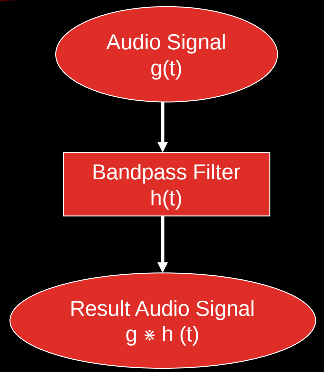


Figure 3.13 Short-time Fourier transform using a long (500 points or 50 msec) Hamming window on a section of unvoiced speech.

From: Rabiner et al.



BAND PASS FILTER

Note that the band pass filter can be defined as:

- a *convolution* with a filter response function h(t) in the time domain
- a *multiplication* with a filter response H(f) function in the frequency domain

$$g * h\,(t) = \int_{-\infty}^{\infty} g(\tau)h(t-\tau)d\tau \leftrightarrow G(f) \cdot H(f)$$

BANK OF FILTERS ANALYSIS MODEL

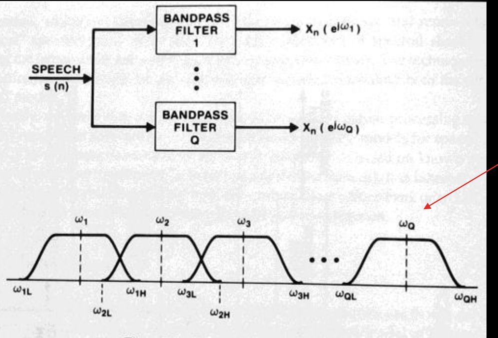


Figure 3.2 Bank-of-filters analysis model.

Central Frequency

MEL-CEPSTRUM [4]

Auditory characteristics

- Mel-scaled filter banks

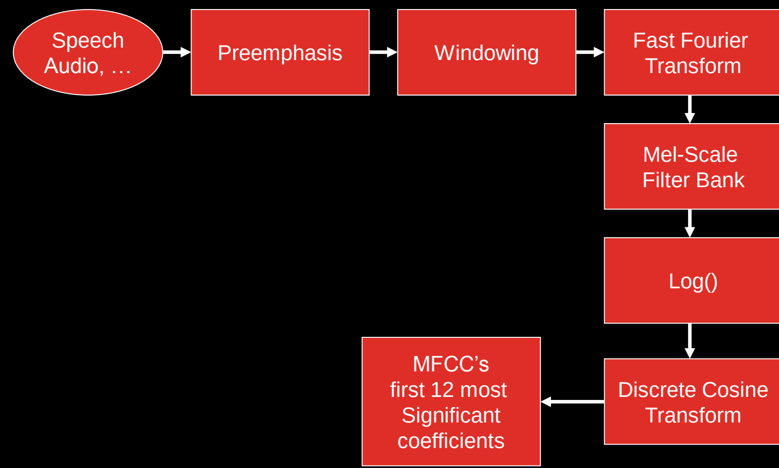
De-correlating properties

- by applying a discrete cosine transform (which is close to a Karhunen-Loeve transform) a de-correlation of the mel-scale filter log-energies results
- => probabilistic modeling on these de-correlated coefficients will be more effective.

One of the most successful features for speech recognition, speaker recognition, and other speech related recognition tasks.

[1, pp 712-717]

MFCCS



Automatic Speech Recognition Architectures
Incorporating Multiple Knowledge Sources

Input Speech

Acoustic Front-end

Acoustic Models P(A/W)

Language Model P(W)

Search

Recognized Utterance

- The signal is converted to a sequence of feature vectors (spectral and temporal).
- Acoustic models represent sub-word units, such as phonemes: finite-state machine models spectral structure and temporal structure.
- The language model predicts the next set of words, and controls which models are hypothesized. (N-grams)
- Search to find the most probable word sequence.

Acoustic Modeling
Feature Extraction

Input Speech

Fourier Transform

Cepstral Analysis

Perceptual Weighting

Time Derivative

Time Derivative

Energy + Mel-Spaced Cepstrum

Delta Energy + Delta Cepstrum

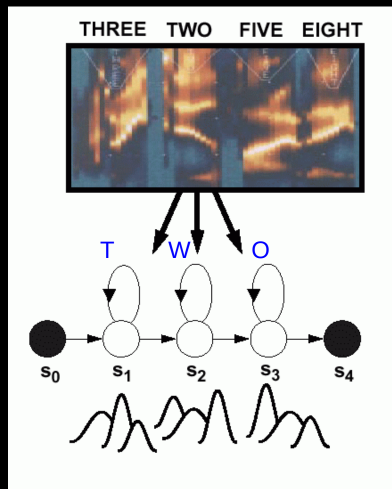
Delta-Delta Energy + Delta-Delta Cepstrum

- Incorporate knowledge of the nature of speech sounds in measurement of the features.
- Utilize rudimentary models of human perception.

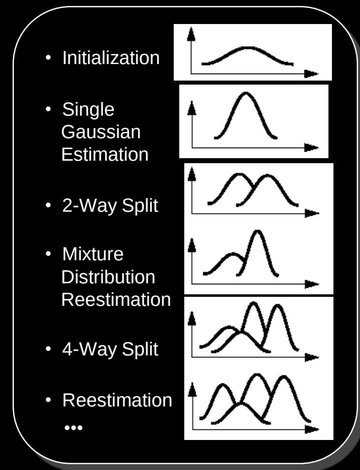
- Typically: 512 samples (16kHz sampling rate) =>
- Use a ~30 msec window for frequency domain analysis.
- Include absolute energy and 12 spectral measurements.
- Time derivatives to model spectral change.

Acoustic Modeling
Hidden Markov Models

- Acoustic models: temporal evolution of the features (spectrum).
- Gaussian mixture distributions for variations in speaker, accent, and pronunciation.
- Phonetic model topologies are simple left-to-right structures.
- Skip states (time-warping) and multiple paths (alternate pronunciations).
- Sharing model parameters to reduce complexity.



Acoustic Modeling
Parameter Estimation



- Word level transcription
- Supervises a closed-loop data-driven modeling
- Initial parameter estimation
- The expectation/maximization (EM) algorithm is used to improve our parameter estimates.
- Computationally efficient training algorithms (Forward-Backward) are crucial.
- Batch mode parameter updates are typically preferred.
- Decision trees and the use of additional linguistic knowledge are used to optimize parameter-sharing, and system complexity,.

MACHINE LEARNING METHODS

- k Nearest Neighbors
- Decision Trees
- Random Forests (weighted neighborhoods scheme)
- Gradient Boosting Machines (e.g. boosting of prediction model ensembles)
- Vector Quantization
  - Finite code book of spectral shapes
  - The code book codes for 'typical' spectral shape
  - Method for all spectral representations (e.g. Filter Banks, LPC, ZCR, etc. …)
- Support Vector Machines
- Markov Models
- Hidden Markov Models
- Neural Networks Etc.

VECTOR QUANTIZATION

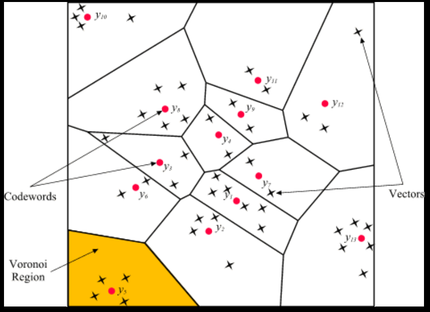
- Data represented as feature vectors.
- Vector Quantization (VQ) Training set => determine a set of code words that constitute a code book.
- Code words are centroids using a similarity or distance measure d.
- Code words together with measure d divide the space into Voronoi regions.
- A query vector falls into a Voronoi region and will be represented by the respective code word.

[2, pp. 466 – 467]

VECTOR QUANTIZATION

Distance measures d(x,y):

- Euclidean distance
- Taxi cab distance
- Hamming distance
- etc.



VECTOR QUANTIZATION

Let a training set of L vectors be given for a certain class of objects.
Assume a codebook of M code words is wanted for this class.

Initialize:

- choose M arbitrary vectors of the L vectors of the training set.
- This is the initial code book.

Nearest Neighbor Search:

- for each training vector v, find the code word w in the current code book that is closest and assign v to the corresponding cell of w.

Centroid Update:

- For each cell with code word w determine the centroid c of the training vectors that are assigned to the cell of w.
- Update the code word w with the new vector c.

Iteration:

- repeat the steps **Nearest Neighbor Search** and **Centroid Update** until the average distance between the new and previous code words falls below a preset threshold.

VECTOR CLASSIFICATION

For an M-vector code book CB with codes

$CB = \{y_i \mid 1 \leq i \leq M\}$,

the index m^* of the best codebook entry for a given vector v is:

$$m^* = \underset{1 \leq i \leq M}{\arg\min}\; d(v, y_i)$$

VQ FOR CLASSIFICATION

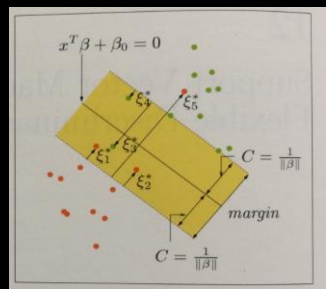
A code book $CB_k = \{y^k_i \mid 1 \leq i \leq M\}$, can be used to define a class C_k.

Example Audio Classification:

- Classes 'crowd', 'car', 'silence', 'scream', 'explosion', etc.
- Determine by using VQ code books CB_k for each of the respective classes C_k.
- VQ is very often used as a baseline method for classification problems.

SUPPORT VECTOR MACHINES

- A generalization of linear decision boundaries for classification.
- Necessary when classes overlap when using linear decision boundaries (non separable classes).



From: [2]

Find hyper plane P: $x^T\beta + \beta_0 = 0$, such that

$\|\beta\|$ is minimized over $\begin{cases} y_i(x_i^T\beta + \beta_0) \geq 1 - \varepsilon_i \ \forall i \\ \varepsilon_i \geq 0, \ \ \sum \varepsilon_i \leq constant \end{cases}$

Where $(x_1,y_1), \ldots (x_N,y_N)$ are our training pairs, with $x_i \in \mathbb{R}^p$ and $y_i \in \{-1,1\}$,

$\varepsilon = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N)$ are the slack variables, i.e.,

ε_i = the amount that x_i is on the wrong side of the margin $C = \frac{1}{\|\beta\|}$ from the hyper plane P.

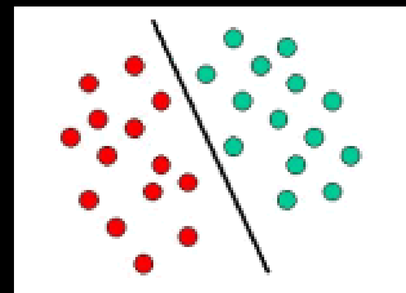
i.e. C is maximized.

=> Problem is quadratic with linear inequalities constraint. [2, pp 377-389]

SUPPORT VECTOR MACHINE (SVM)

In this method so called support vectors define decision boundaries for classification and regression.

An example where a straight line separates the two Classes: a linear classifier



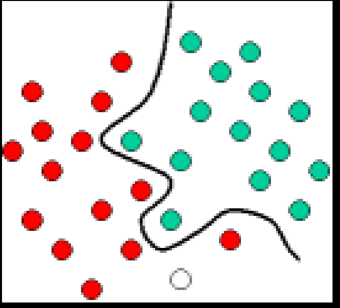
Images from: www.statsoft.com.

SUPPORT VECTOR MACHINE (SVM)

In general classification is not that simple.

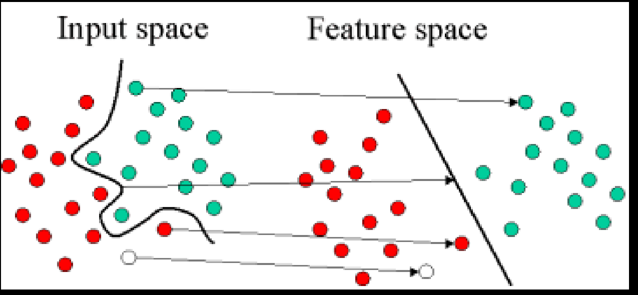
SVM is a method that can handle the more complex cases where the decision boundary requires a curve.

SVM uses a set of mapping functions (kernels) to map the feature space into a transformed space so that hyperplanes can be used for the classification.



SUPPORT VECTOR MACHINE (SVM)

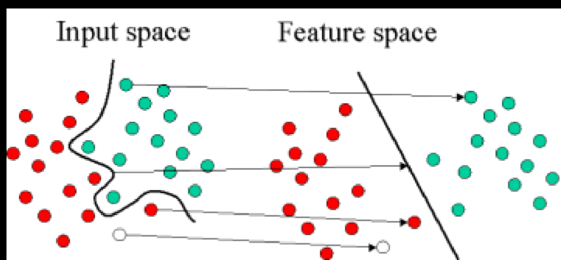
SVM uses a set of mapping functions (kernels) to map the feature space into a transformed space so that hyperplanes can be used for the classification.



SUPPORT VECTOR MACHINE (SVM)

Training of an SVM is an iterative process:

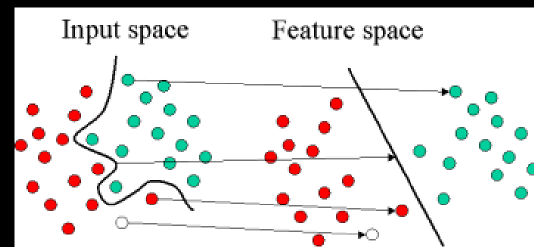
- optimize the mapping function while minimizing an error function
- The error function should capture the penalties for misclassified, i.e., non separable data points.



SUPPORT VECTOR MACHINE (SVM)

SVM uses kernels that define the mapping function used in the method. Kernels can be:

- Linear
- Polynomial
- RBF
- Sigmoid
- Etc.



- RBF (radial basis function) is the most popular kernel, again with different possible base functions.
- The final choice depends on characteristics of the classification task.

AUDIO CLASSIFICATION USING NEURAL NETWORKS

An example by Rishi Sidhu:

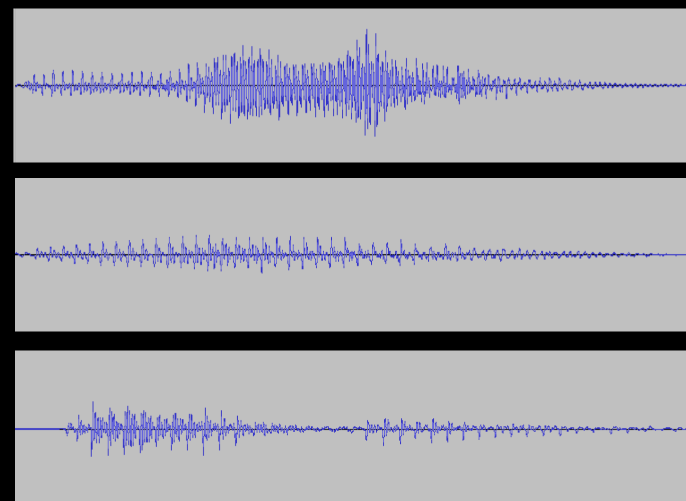
https://medium.com/x8-the-ai-community/audio-classification-using-cnn-coding-example-f9cbd272269e

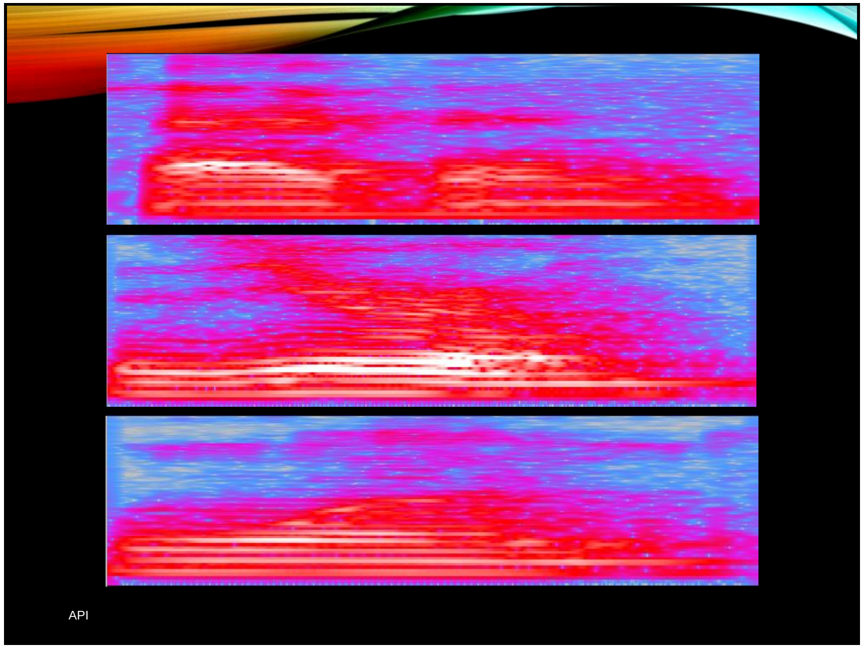
Using data from the **Spoken Digit Dataset** by Zohar Jackson:

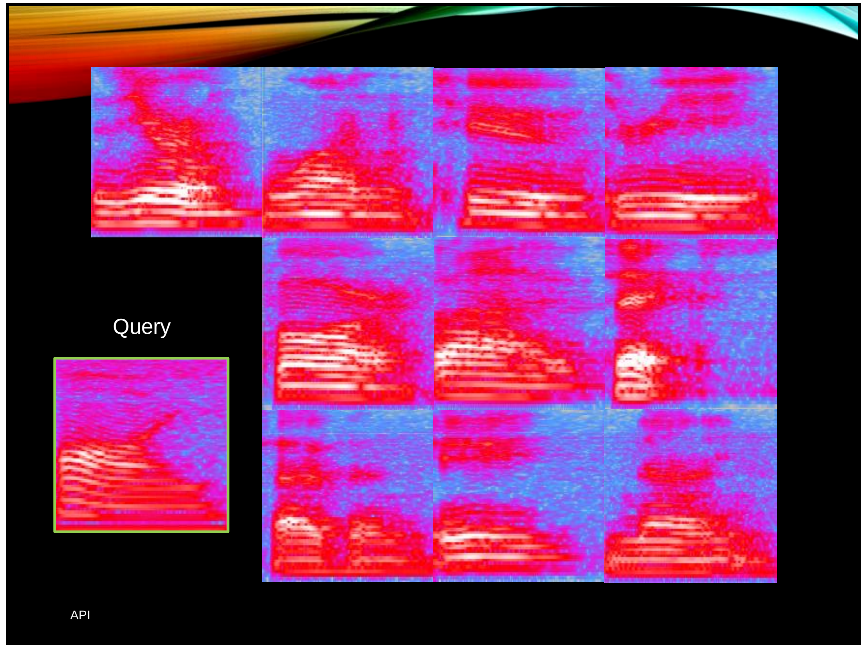
Https://github.com/Jakobovski/free-spoken-digit-dataset

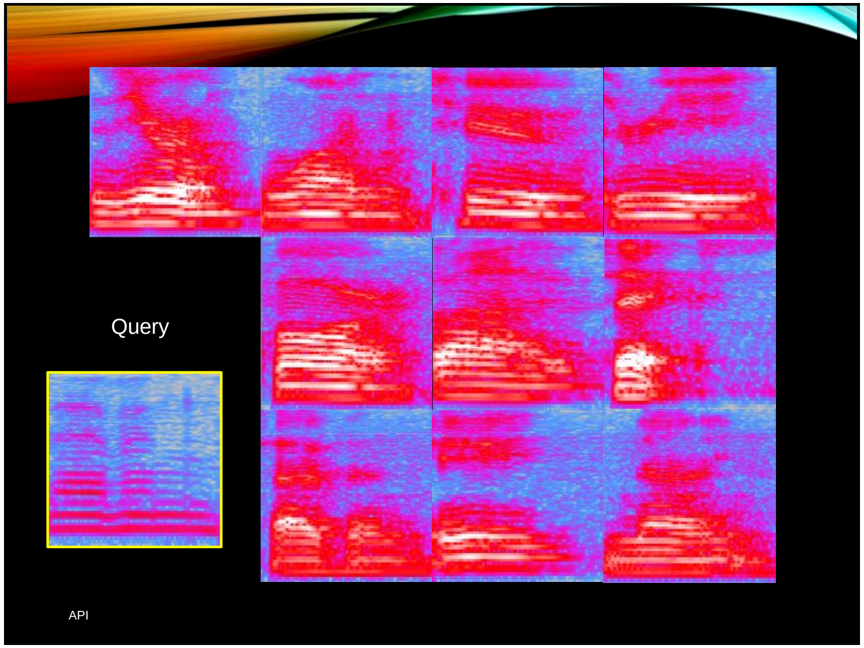
Using Convolutional Neural Networks on Spectograms.

DIGITS

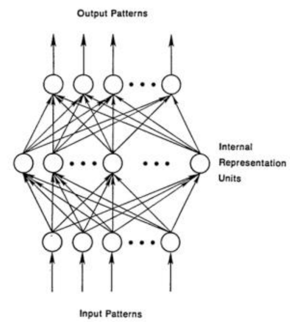



API

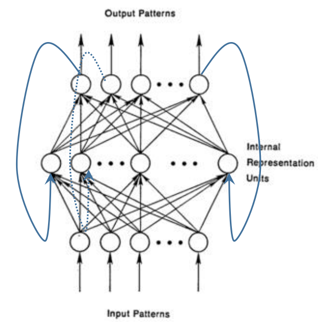

Query
API



Some Neural Networks



Feed Forward Neural Network



Recurrent Neural Network

DNN: AlexNet, VGG16, ResNet, etc.

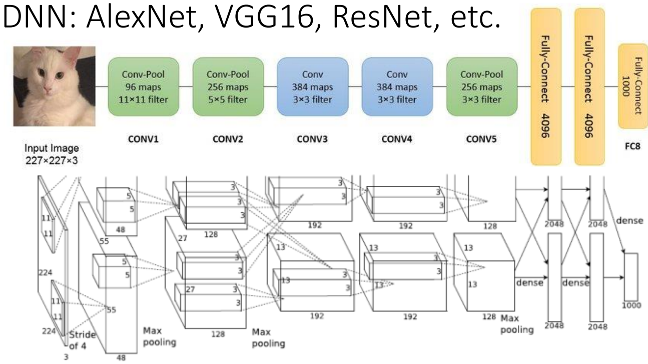


Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. "ImageNet classification with deep convolutional neural networks" Communications of the ACM. 60 (6): 84–90.

Deep Visualization Toolbox

yosinski.com/deepvis

#deepvis

Cornell University

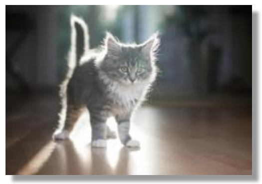
UNIVERSITY OF WYOMING

Jet Propulsion Laboratory
California Institute of Technology

Cats and Dogs

Kaggle Dataset (https://www.kaggle.com/c/dogs-vs-cats/data)

- 2000 images of cats
- 2000 images of dogs

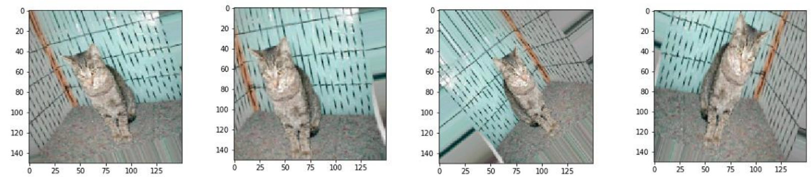


- Given an image: is it a cat or a dog?

Divide into:

- Training set (2000 images)
- Validation set (1000 images)
- Test set (1000 images)

Cats and Dogs

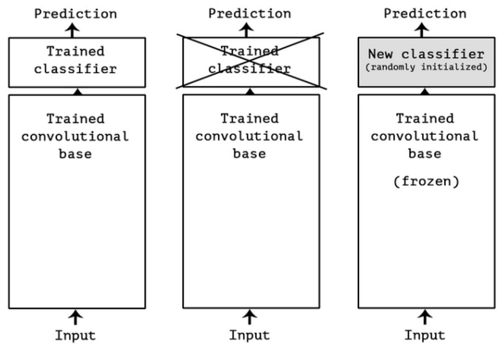


Convolutional Neural Network

- Without any regularization: ~71% accuracy
- With data augmentation: ~82% accuracy
- Feature extraction using a pre-trained NN: ~90% accuracy
- Fine tuning a pre-trained NN: ~95% accuracy

These are examples of Deep Learning with Small Datasets.

Cats and Dogs



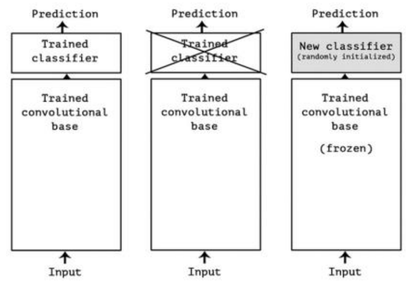
VGG16 (pre packed with Keras)

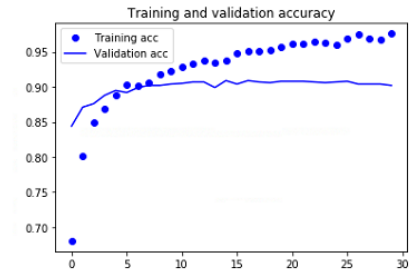
Convolutional Neural Network

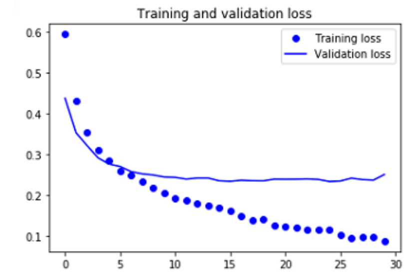
- Without any regularization: ~71% accuracy
- With data augmentation: ~82% accuracy
- Feature extraction using a pre-trained NN: ~90% accuracy
- Fine tuning a pre-trained NN: ~95% accuracy

These are examples of Deep Learning with Small Datasets.

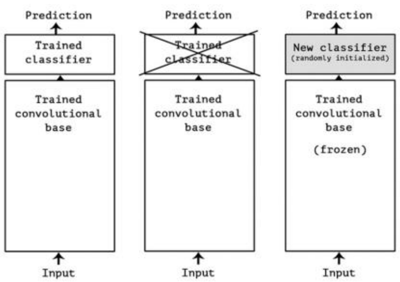
VGG16
Feature Extraction

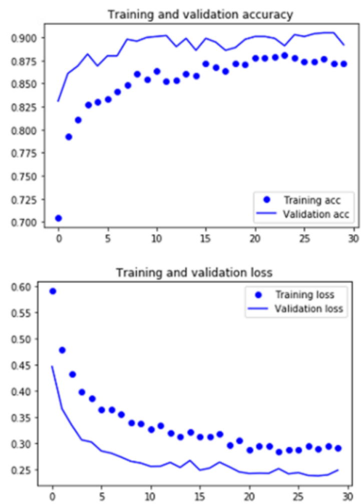






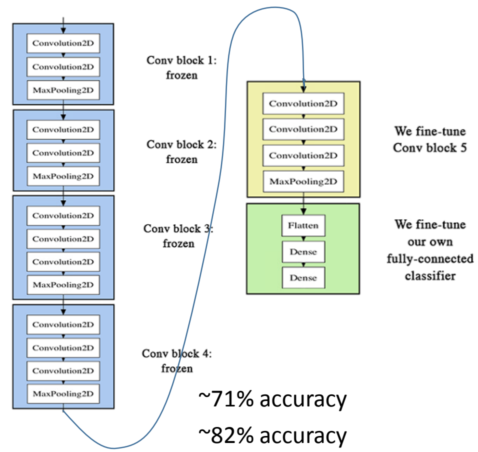
VGG16
Feature Extraction +
Data Augmentation





Cats and Dogs

VGG16 (pre packed with Keras)

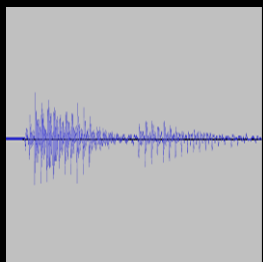


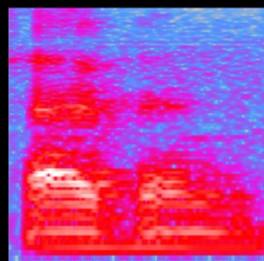
Convolutional Neural Network

- Without any regularization: ~71% accuracy
- With data augmentation: ~82% accuracy
- Feature extraction using a pre-trained NN: ~90% accuracy
- Fine tuning a pre-trained NN: ~95% accuracy

These are examples of Deep Learning with Small Datasets.

CNN'S FOR AUDIO CLASSIFICATION





- Both images can be used to recognize the spoken digit.
- The spectrogram yields better accuracy for the tests.

CNN ARCHITECTURE

Input Layer
↓
Convolutional layer with kernel size 3x3
↓
Convolutional layer with kernel size 3x3
↓
Max Pooling layer with pool size 2x2
↓
Dropout layer
↓
Flattening layer
↓
Dense layer 1
↓
Dense layer 1

CNN DEFINED IN TF.KERAS

```
#Define Model
model = Sequential()
model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=input_shape))
model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax'))
#Compile
model.compile(loss=keras.losses.categorical_crossentropy,
optimizer=keras.optimizers.adam(), metrics=['accuracy'])
print(model.summary())
#Train and Test The Model
model.fit(x_train, y_train, batch_size=4, epochs=10, verbose=1, validation_data=(x_test,
y_test))
```


TRAINING, TEST AND VALIDATION DATASETS

Training Data

- 1800 Images of Spectrograms: 34x50 pixels
- Each image is labeled with the correct digit

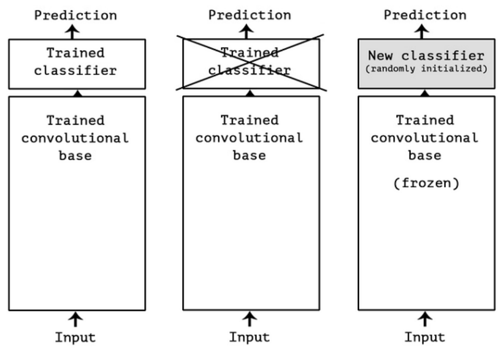
Validation Data

- 200 Images of Spectrograms: 34x50 pixels
- Each image is labeled with the correct digit
- Exclusive speaker(s)

Test Data

- 200 Images of Spectrograms: 34x50 pixels
- Each image is labeled with the correct digit
- Exclusive speaker(s)

Digits



VGG16 (pre packed with Keras)

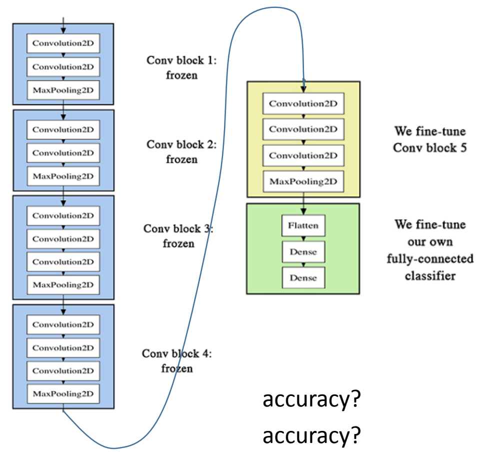
Convolutional Neural Network

- Without any regularization: accuracy ?
- With data augmentation: accuracy ?
- Feature extraction using a pre-trained NN: accuracy ?
- Fine tuning a pre-trained NN: accuracy ?

These are examples of Deep Learning with Small Datasets.

Digits

VGG16 (pre packed with Keras)



Convolutional Neural Network

- Without any regularization: accuracy?
- With data augmentation: accuracy?
- Feature extraction using a pre-trained NN: accuracy?
- Fine tuning a pre-trained NN: accuracy?

These are examples of Deep Learning with Small Datasets.

W. Chunyang et al. Transformer-based Acoustic Modeling for Streaming Speech Synthesis, INTERSPEECH 2021

https://transformer-tts-accoustic-model.github.io/samples/

Tacotron2 uses Bi-directional Long Short-term Memory (BLSTM) recurrent networks.

- cannot effectively model long-term dependencies
- a poor quality on long speech.

FastSpeech state-of-the-art

- in modeling speech prosody and spectral features, but
- computation is parallel over the full utterance context.

W. Chunyang et al. Transformer-based Acoustic Modeling for Streaming Speech Synthesis, INTERSPEECH 2021

TTS systems usually consist of two stages:

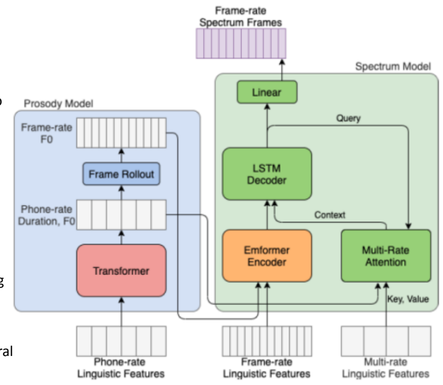
- acoustic model that predicts the prosody and spectral features
- followed by a neural vocoder that generates the audio
- waveform.

Tranformer models:

- model long-term dependencies
- Complexity grows quadratically

This work

- Effcient constant speed implementation: for streaming speech synthesis
- uses a transformer network that predicts the prosody features at phone rate
- an Emformer network to predict the frame-rate spectral features (streaming)
- WaveRNN Vocoder used



https://transformer-tts-accoustic-model.github.io/samples/

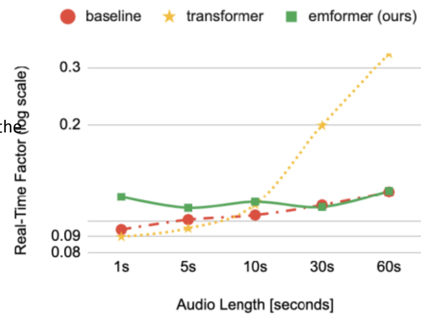
W. Chunyang et al. Transformer-based Acoustic Modeling for Streaming Speech Synthesis, INTERSPEECH 2021

TTS systems usually consist of two stages:

- acoustic model that predicts the prosody and spectral features
- followed by a neural vocoder that generates the audio
- waveform.

Tranformer models:

- model long-term dependencies
- Complexity grows quadratically



System	Prosody	Spectrum	Normal	Long
Groundtruth	–	–	4.307 ± 0.037	4.360 ± 0.044
Baseline [11]	BLSTM with self-attention [26]	Multi-rate attention [11]	4.173 ± 0.042	4.019 ± 0.055
Ours-1	Transformer	Multi-rate attention	4.174 ± 0.042	4.107 ± 0.052
Ours-2	BLSTM with self-attention	Emformer with multi-rate attention	4.192 ± 0.041	4.034 ± 0.053
Ours-3 (best)	Transformer	Emformer with multi-rate attention	**4.213 ± 0.042**	**4.201 ± 0.048**

https://transformer-tts-accoustic-model.github.io/samples/

REFERENCES

1. T.F. Quatieri, Discrete-Time Speech Signal Processing, Principles and Practice, Prentice-Hall, Inc. 2002.
2. T. Hastc, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Data Mining, Inference, and Prediction, Springer, 2001.
3. W.H. Press, S.A.Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipies in C++, The Art of Scientific Computing, 2nd Edition, Cambridge University Press, 2002.
4. S.B. Davies, P. Mermelstein, Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences, IEEE Trans. Acoustics, Speech, and Signal Processing, vol. ASSP-28, no.4, pp. 357-366, Aug. 1980.

API

REFERENCES

5. P. Kenny, "Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms, Tech. Report CRIM-06/08-13," 2005.

Available: http://www.crim.ca/perso/patrick.kenny

6. N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Frontend factor analysis for speaker verification," IEEE Trans. Audio, Speech, Lang. Process., vol. 19, no. 4, pp. 788–798, May 2011.

7. François Chollet, Deep Learning with Python, Manning Publications, November 2017.

Name	Family	Session	Title
Joana	Trashlieva	1	P. Ansar Khan et al., AUDIO SENTIMENT ANALYSIS. IJCRT, 2021.
Ruilin	Wang	1	I. Baumann et al., Nonwords Pronunciation Classification in Language Development Tests fro Preschool Children. Interspeech 2022.
Shihang	YU	1	L. Nanni et al., Ensemble of convolutional neural networks to improve animal audio classification. EURASIP J. Audio Speech Music. Process., 2020.
Tushar	Pal	1	A.L. Wang, An Industrial-Strength Audio Search Algorithm. Shazam Entertainment, Ltd. Xxxx.
Victor	Batenburg	1	S. Hantke et al., WHAT IS MY DOG TRYING TO TELL ME? THE AUTOMATIC RECOGNITION OF THE CONTEXT AND PERCEIVED EMOTION OF DOG BARKS. Xxxx.
Wei	Chen	1	B. McFee et al., LARGE-SCALE MUSIC SIMILARITY SEARCH WITH SPATIAL TREES. ISMIR 2011.
Xinyue	Xie	1	J.M. Arend et al., Efficient binaural rendering of spherical microphone array data by linear filtering. EURASIP J. Audio Speech Music. Process., 2021.
Yu	Ang_Yuan	1	F. Lluis et al., End-to-end music source separation: is it possible in the waveform domain? Interspeech 2019.
Diego	Barreiro_Clemente	2	Chunyan Ji, et al., A review of infant cry analysis and classification. EURASIP J. Audio Speech Music. Process. 2021.
Haoran	Yin	2	R. Hebbar et al., Deep multiple instance learning for foreground speech localization in ambient audio from wearable devices. EURASIP J. Audio Speech Music. Process., 2021.
Hielke	Muizelaar	2	H. Yakura et al., Self-Supervised Contrastive Learning for Singing Voices. IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 30, 2022/
Saleh	Saleh_Tarek_Issa_Alwer	2	A. Radford et al., Robust Speech Recognition via Large-ScaleWeak Supervision. Xxxx.
Shubham	Bhatt	2	C. Brazier et al., ON-LINE AUDIO-TO-LYRICS ALIGNMENT BASED ON A REFERENCE PERFORMANCE. ISMIR2021.
Sudarshna	Arya_Patel	2	L. Lin et al. A UNIFIED MODEL FOR ZERO-SHOT MUSIC SOURCE SEPARATION, TRANSCRIPTION AND SYNTHESIS. ISMIR2021.
Yijie	Lu	2	L.-C. Chang et al., MIDINET: A CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORK FOR SYMBOLIC-DOMAIN MUSIC GENERATION. ISMIR2017.
ZHU	OU	2	T. Tse et al., ENSEMBLE: A HYBRID HUMAN-MACHINE SYSTEM FOR GENERATING MELODY SCORES FROM AUDIO. ISMIR2016.
Aria	Tian	3	S. Chowdhury et al., ON PERCEIVED EMOTION IN EXPRESSIVE PIANO PERFORMANCE: FURTHER EXPERIMENTAL EVIDENCE FOR THE RELEVANCE OF MID-LEVEL PERCEPTUAL FEATURES,. SMIR2021.
George	Boukouvalas	3	L. Tian et al., RECOGNIZING EMOTIONS IN SPOKEN DIALOGUE WITH HIERARCHICALLY FUSED ACOUSTIC AND LEXICAL FEATURES. xxxx.
Jesse	de_Gans	3	C. Mitcheltree et al., White-box Audio VST Effect Programming, Feb 2021.
Junduo	Sun	3	J. Gao et al., Black-box adversarial attacks through speech distortion for speech emotion recognition. EURASIP J. Audio Speech Music. Process., 2022.
Juri	Morisse	3	S. Kahl et al., BirdNET: A deep learning solution for avian diversity monitoring. Ecological Informatics, 2021.
Lin	He	3	J. Lee, The Emotion is Not One-Hot Encoding: Learning with Grayscale Label for Emotion Recognition in Conversation. Interspeech 2022.
Priyansh	Jain	3	K. Choi et al., LISTEN, READ AND IDENTIFY: MULTIMODAL SINGING LANGUAGE IDENTIFICATION OF MUSIC. ISMIR2021.
RUILIN	MA	3	H. Ankishan, Blood pressure prediction from speech recordings. Biomedical Signal Processing and Control, 2020.
Felicia	Redelaar	4	A. Wiggins et al., GUITAR TABLATURE ESTIMATION WITH A CONVOLUTIONAL NEURAL NETWORK. ISMIR2019.
Luuk	van_den_Nouweland	4	D. Kaminska et al., Efficiency of chosen speech descriptors in relation to emotion recognition. EURASIP J. Audio Speech Music. Process., 2017.
Marc	Boel	4	F. Nadeem, LEARNING FROM MUSICAL FEEDBACK WITH SONG THE HEDGEHOG. SMIR2021.
Matt	van_den_Nieuwenhuijzen	4	V. Madaghiele et al., MINGUS: MELODIC IMPROVISATION NEURAL GENERATOR USINGSEQ2SEQ, SMIR2021.
Matthijs	de_Zeeuw	4	Y. Xiang et al., A speech enhancement algorithm based on a non-negative HMM and Kullback-Leibler divergence. EURASIP J. Audio Speech Music. Process., 2022.
Michael	de_Rooij	4	A. Lerch et al., MUSIC PERFORMANCE ANALYSIS: A SURVEY, ISMIR2019.
Pim	Bax	4	M. He at al., Neural Lexicon Reader: Reduce Pronunciation Errors in End-to-end TTS by Leveraging External Textual Knowledge, Interspeech 2022
Sava	Sharif	4	M. Sarfati et al., COMMUNITY-BASED COVER SONG DETECTION. ISMIR2019.
Wenqian	Hu	4	Y. Wang et al., Active Few-Shot Learning for Sound Event Detection. Xxxx.

Name	Family	Session	Title
Joana	Trashlieva	1	P. Ansar Khan et al., AUDIO SENTIMENT ANALYSIS. IJCRT, 2021.
Ruilin	Wang	1	I. Baumann et al., Nonwords Pronunciation Classification in Language Development Tests fro Preschool Ch
Shihang	YU	1	L. Nanni et al., Ensemble of convolutional neural networks to improve animal audio classification. EURASI
Tushar	Pal	1	A.L. Wang, An Industrial-Strength Audio Search Algorithm. Shazam Entertainment, Ltd. Xxxx.
Victor	Batenburg	1	S. Hantke et al., WHAT IS MY DOG TRYING TO TELL ME? THE AUTOMATIC RECOGNITION OF THE CONTEXT AN
Wei	Chen	1	B. McFee et al., LARGE-SCALE MUSIC SIMILARITY SEARCH WITH SPATIAL TREES. ISMIR 2011.
Xinyue	Xie	1	J.M. Arend et al., Efficient binaural rendering of spherical microphone array data by linear filtering. EURAS
Yu	Ang_Yuan	1	F. Lluis et al., End-to-end music source separation: is it possible in the waveform domain? Interspeech 201
Diego	Barreiro_Clemente	2	Chunyan Ji, et al., A review of infant cry analysis and classification. EURASIP J. Audio Speech Music. Process
Haoran	Yin	2	R. Hebbar et al., Deep multiple instance learning for foreground speech localization in ambient audio from
Hielke	Muizelaar	2	H. Yakura et al., Self-Supervised Contrastive Learning for Singing Voices. IEEE/ACM TRANSACTIONS ON AU
Saleh	Saleh_Tarek_Issa_Alwer	2	A. Radford et al., Robust Speech Recognition via Large-ScaleWeak Supervision. Xxxx.
Shubham	Bhatt	2	C. Brazier et al., ON-LINE AUDIO-TO-LYRICS ALIGNMENT BASED ON A REFERENCE PERFORMANCE. ISMIR202
Sudarshna	Arya_Patel	2	L. Lin et al. A UNIFIED MODEL FOR ZERO-SHOT MUSIC SOURCE SEPARATION, TRANSCRIPTION AND SYNTHES
Yijie	Lu	2	L.-C. Chang et al., MIDINET: A CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORK FOR SYMBOLIC-DOM
ZHU	OU	2	T. Tse et al., ENSEMBLE: A HYBRID HUMAN-MACHINE SYSTEM FOR GENERATING MELODY SCORES FROM AUD
Aria	Tian	3	S. Chowdhury et al., ON PERCEIVED EMOTION IN EXPRESSIVE PIANO PERFORMANCE: FURTHER EXPERIMENT
George	Boukouvalas	3	L. Tian et al., RECOGNIZING EMOTIONS IN SPOKEN DIALOGUE WITH HIERARCHICALLY FUSED ACOUSTIC AND
Jesse	de_Gans	3	C. Mitcheltree et al., White-box Audio VST Effect Programming, Feb 2021.
Junduo	Sun	3	J. Gao et al., Black-box adversarial attacks through speech distortion for speech emotion recognition. EURA
Juri	Morisse	3	S. Kahl et al., BirdNET: A deep learning solution for avian diversity monitoring. Ecological Informatics, 2021
Lin	He	3	J. Lee, The Emotion is Not One-Hot Encoding: Learning with Grayscale Label for Emotion Recognition in Co
Priyansh	Jain	3	K. Choi et al., LISTEN, READ AND IDENTIFY: MULTIMODAL SINGING LANGUAGE IDENTIFICATION OF MUSIC. I
RUILIN	MA	3	H. Ankishan, Blood pressure prediction from speech recordings. Biomedical Signal Processing and Control,
Felicia	Redelaar	4	A. Wiggins et al., GUITAR TABLATURE ESTIMATION WITH A CONVOLUTIONAL NEURAL NETWORK. ISMIR2019
Luuk	van_den_Nouweland	4	D. Kaminska et al., Efficiency of chosen speech descriptors in relation to emotion recognition. EURASIP J. A
Marc	Boel	4	F. Nadeem, LEARNING FROM MUSICAL FEEDBACK WITH SONG THE HEDGEHOG. SMIR2021.
Matt	van_den_Nieuwenhuijzen	4	V. Madaghiele et al., MINGUS: MELODIC IMPROVISATION NEURAL GENERATOR USINGSEQ2SEQ, SMIR2021.
Matthijs	de_Zeeuw	4	Y. Xiang et al., A speech enhancement algorithm based on a non-negative HMM and Kullback-Leibler dive
Michael	de_Rooij	4	A. Lerch et al., MUSIC PERFORMANCE ANALYSIS: A SURVEY, ISMIR2019.
Pim	Bax	4	M. He at al., Neural Lexicon Reader: Reduce Pronunciation Errors in End-to-end TTS by Leveraging External
Sava	Sharif	4	M. Sarfati et al., COMMUNITY-BASED COVER SONG DETECTION. ISMIR2019.
Wenqian	Hu	4	Y. Wang et al., Active Few-Shot Learning for Sound Event Detection. Xxxx.