Karlsruher Institut für Technologie
Institut für Anthropomatik

Grundlagen der Automatischen Spracherkennung

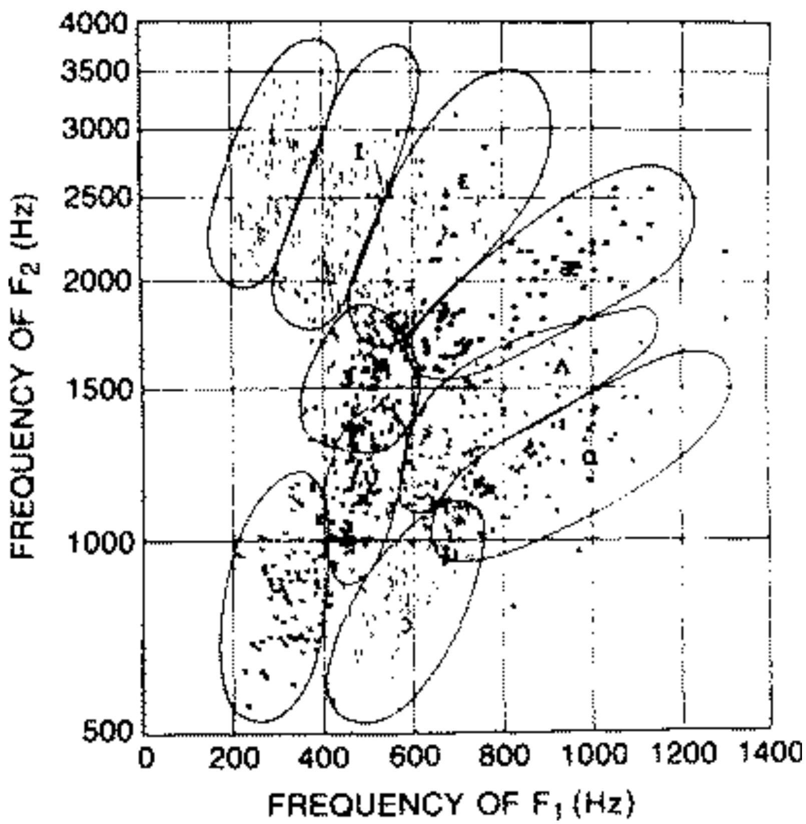
Neuronale Netze

18.1.2012

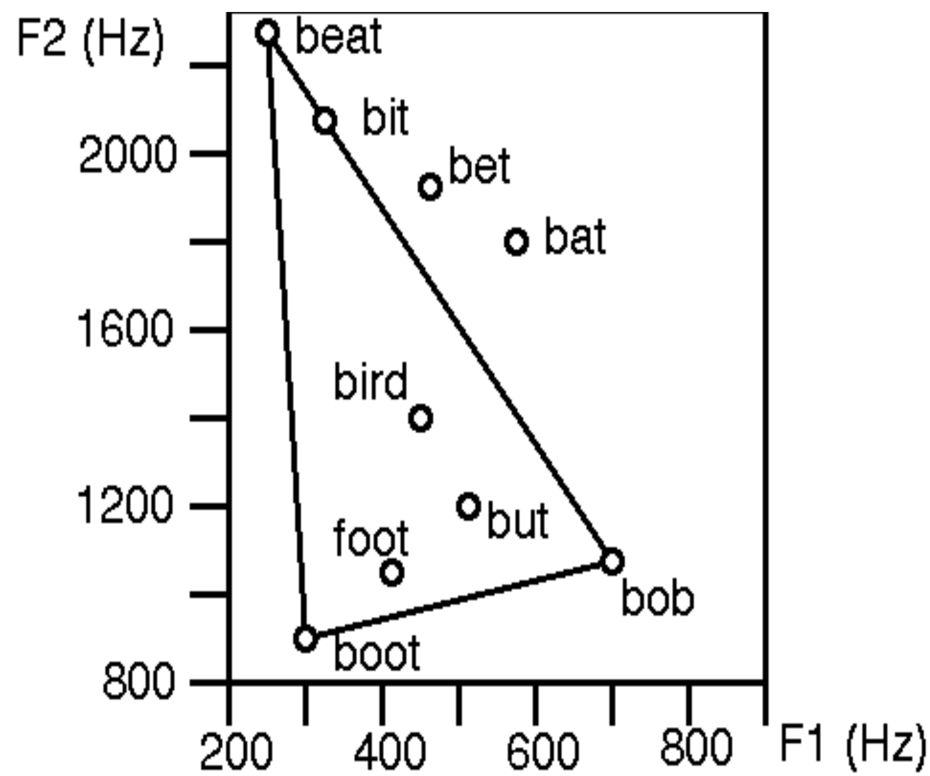
Interactive Systems Labs

Formants

The resonance frequencies of the vocal tract transfer function are called formants. In practice, only the first few formants are of interest.



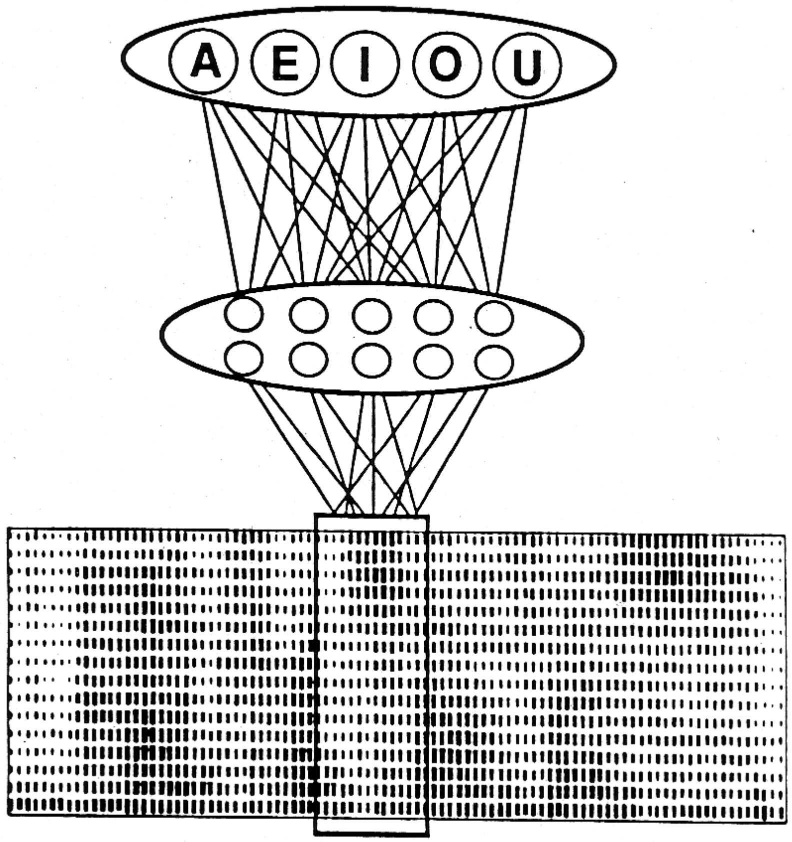
The Vowel-Triangle



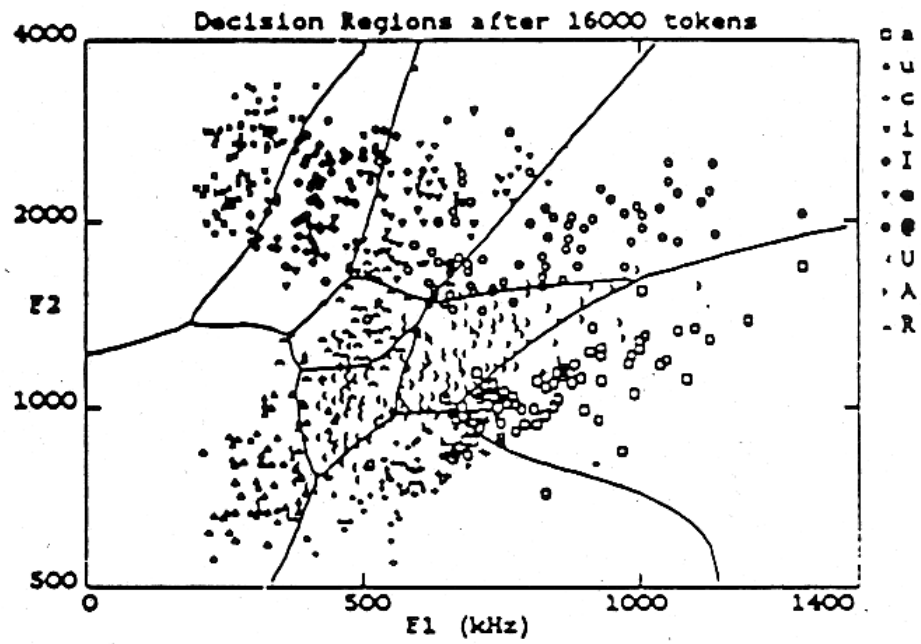
Neural Net Classifiers

- Back-Propagation, Multilayer Perceptrons
- Boltzman Machines
- Decision Tree Classifiers
- Restricted Coulomb Energy
- Feature Map Classifiers
- LVQ, LVQ2
- High Order Networks
- Radial Basis Functions
- Modified Nearest Neighbor

Phoneme Recognition by Classification



Lippmann, Vowel Classification



Phoneme Models

Tasks

- Speaker Independence - Fast and Slow Adaptation
- Continuous Speech
- Phoneme Spotting

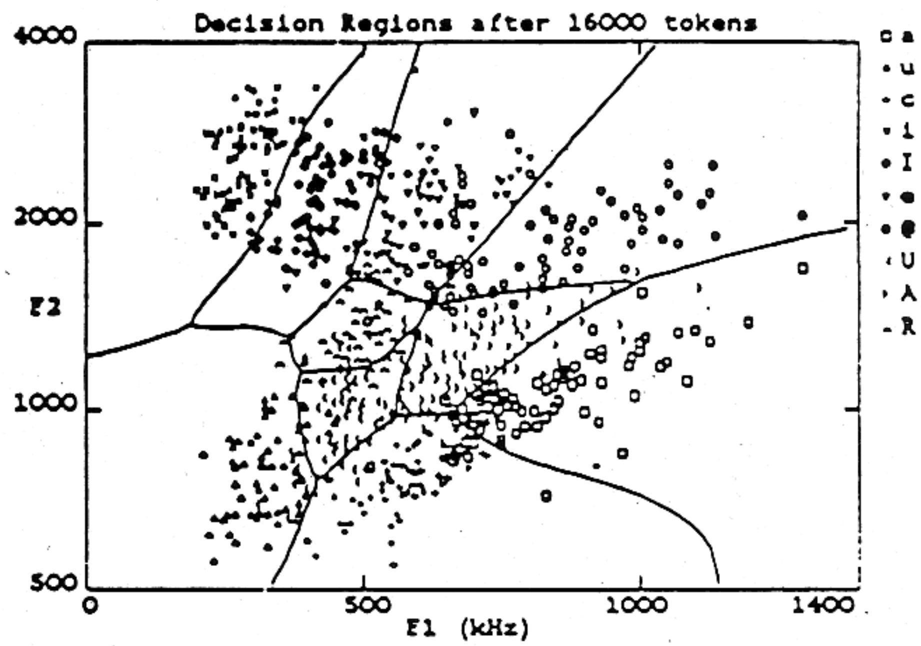
Research Questions

- Objective Functions - Probabilistic Outputs
  and Improved Classification Rate
- Modular Nets, Connectionist Glue, Multiplicative Units
- Adaptive Time-Delays
- Minimal Nets
- Predictive Nets
- Recurrent Nets

Design Criteria

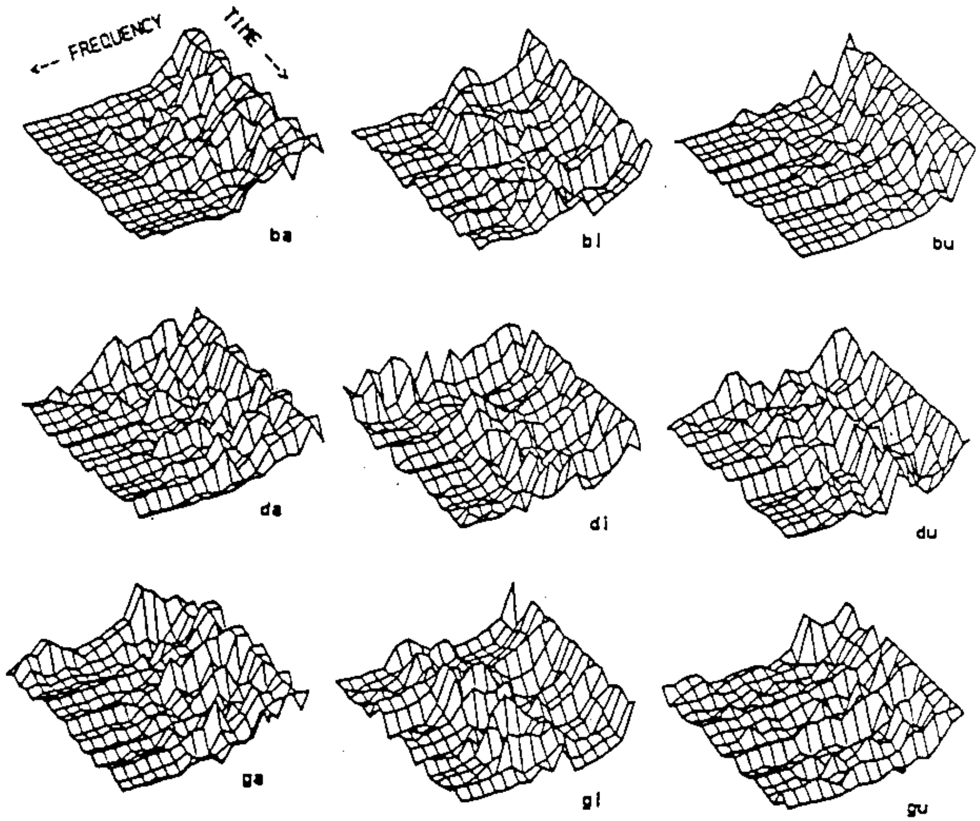
- Recognition Error Rate
- Training Time
- Recognition Time
- Memory Requirements
- Training Complexity
- Ease of Implementation
- Ease of Adaptation

Lippmann, Vowel Classification



Decision regions formed after 16000 training examples
from Peterson and Barney's data. Data samples are also shown.
The Legend shows vowels Arpabet notation.

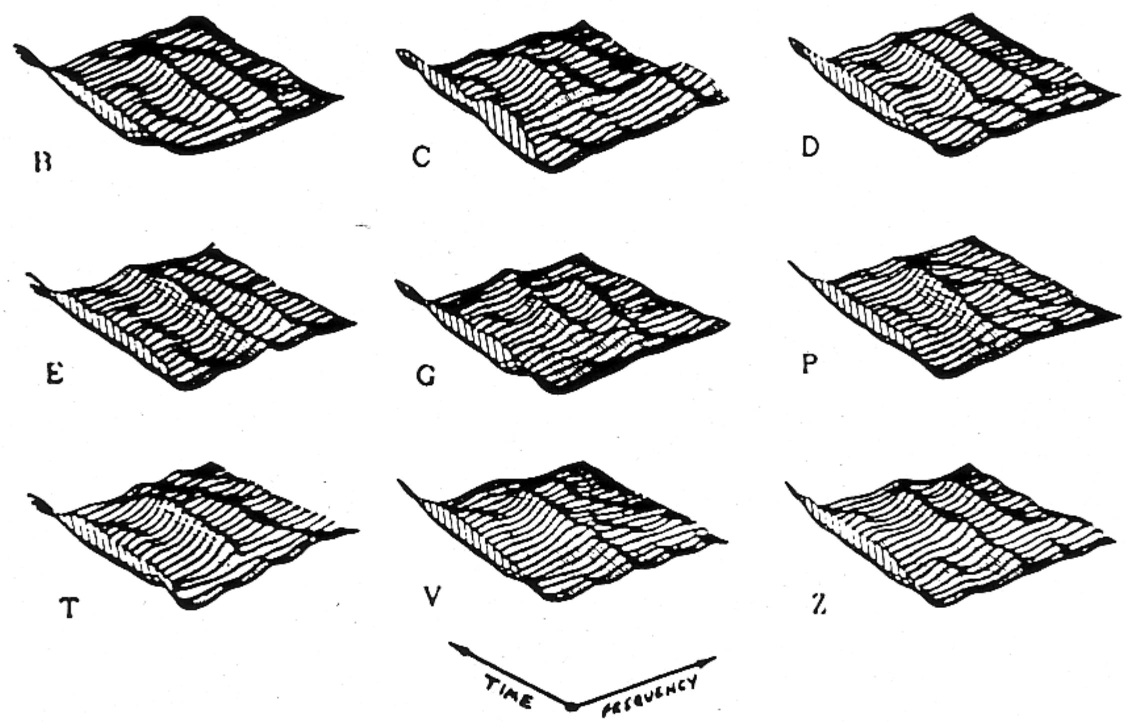
Elman, Vowels, Voiced Stops



Graphs of examples of the nine sounds
[ba], [bi], [bu], [da], [di], [ga], [gi], [gu].

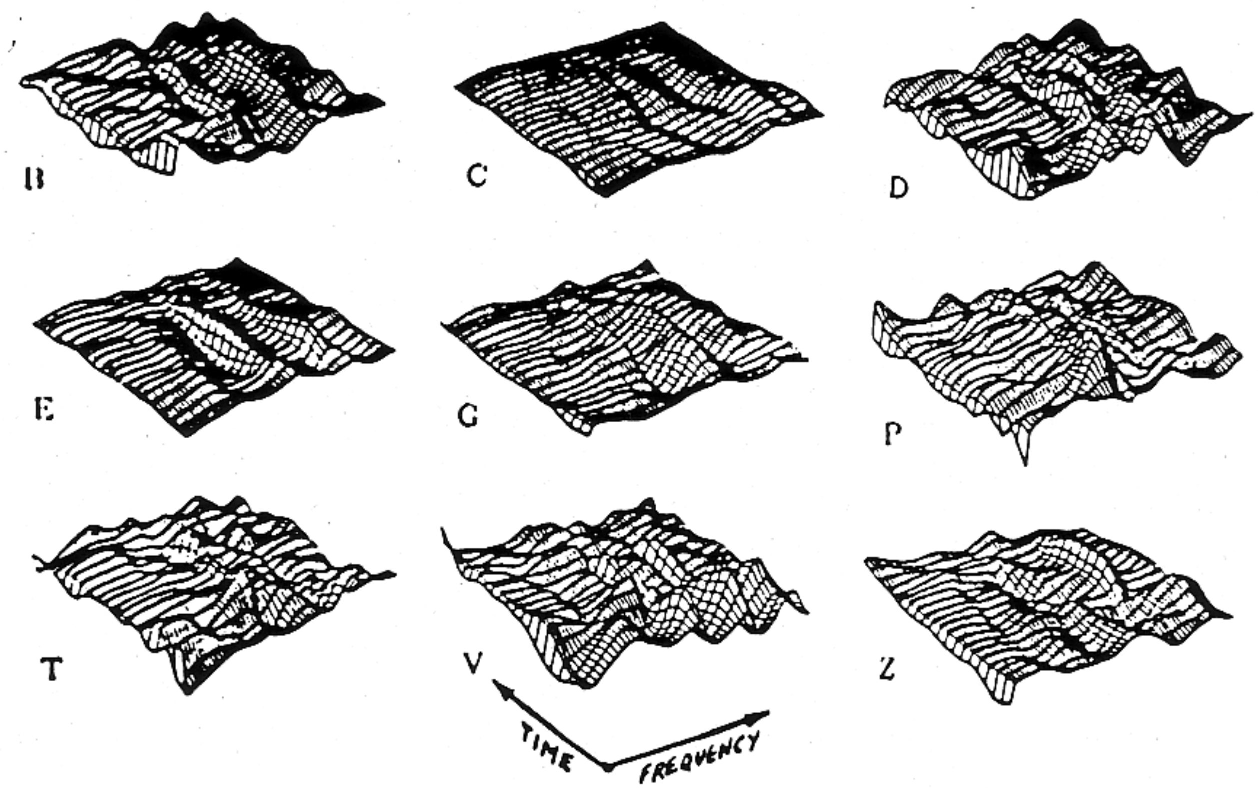
Karlsruher Institut für Technologie

LPC time-frequency plots

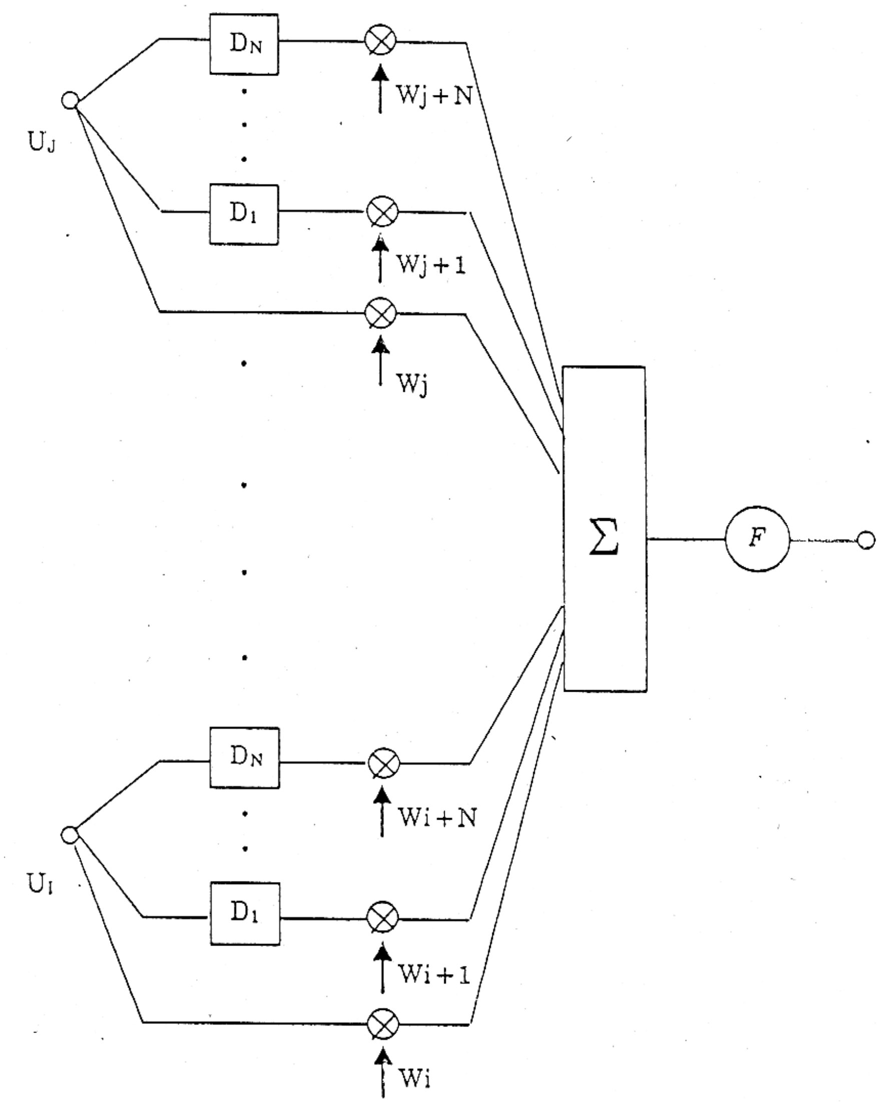


LPC time-frequency plots for representative tokens of the E-set words.

Time-frequency plots (cont.)



Time-frequency plots of weight values connected to each output neuron “B” through “Z” in a trained perceptron.

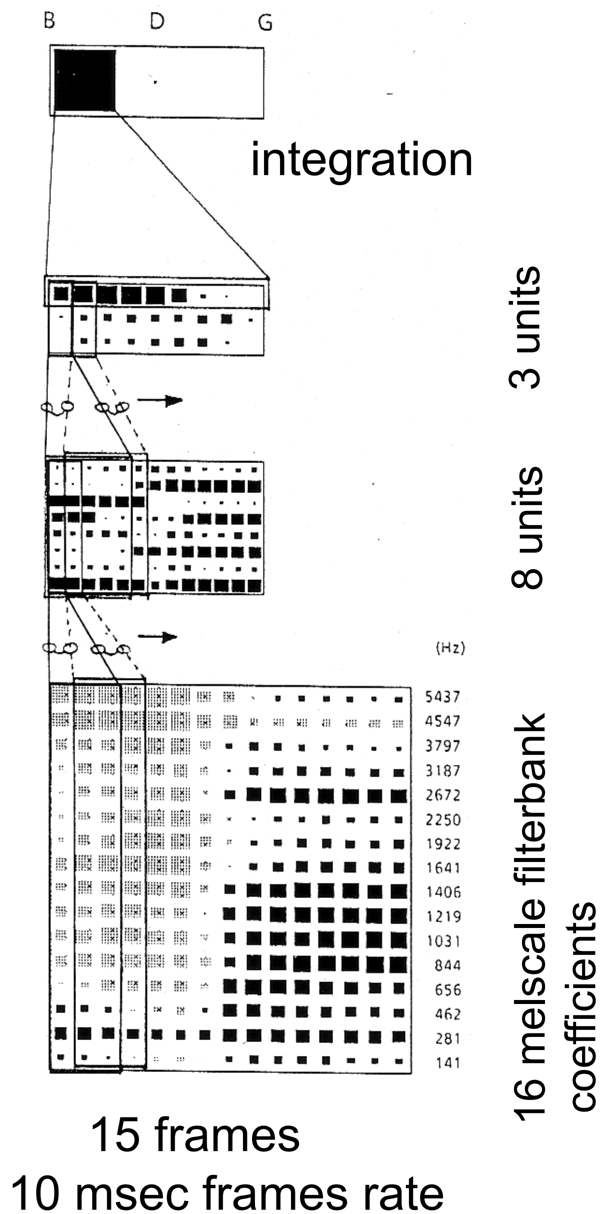


Karlsruher Institut für Technologie

Time-Delay Neural Network

- Multilayer Neural Network - nonlinear decision surfaces
- An appropriate architecture - Integration of speech knowledge. Minimize learning time and amount of training data
- Time-Delay Arrangement - Networks can represent temporal structure of speech
- Translation-Invariant Learning - Hidden units of the network learn features independent of precise location in time

  -> Freedom from precise alignment of segmentation



Output Layer

Hidden Layer 1

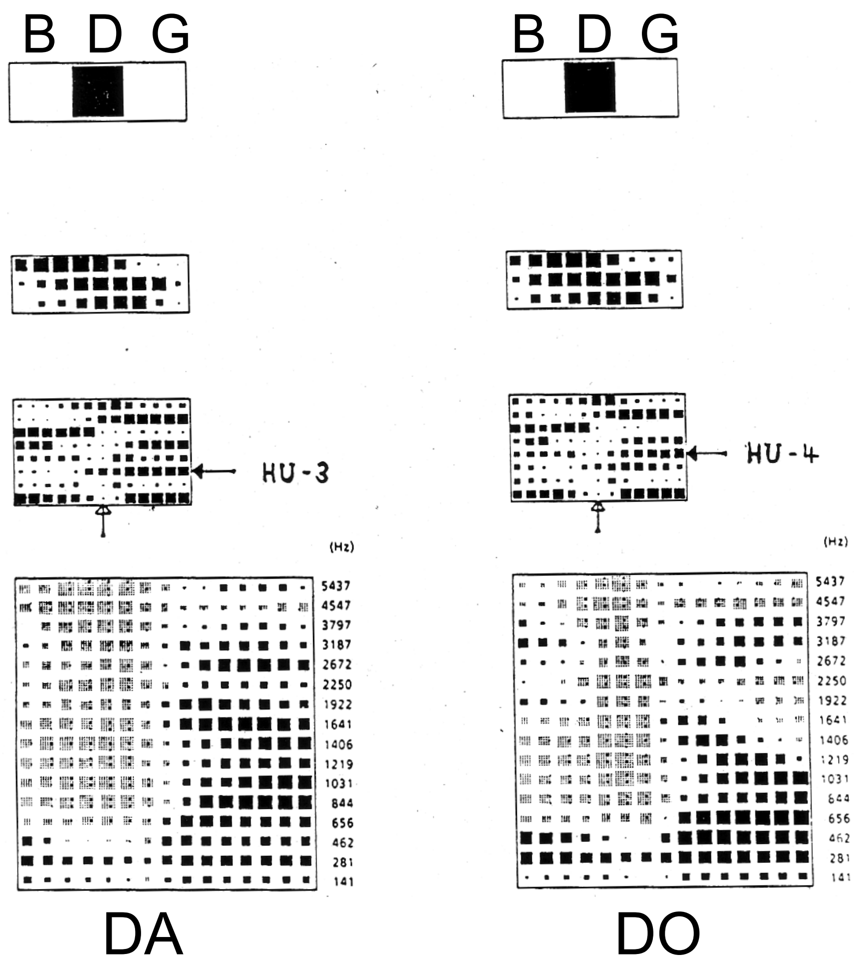
Hidden Layer 2

Input Layer

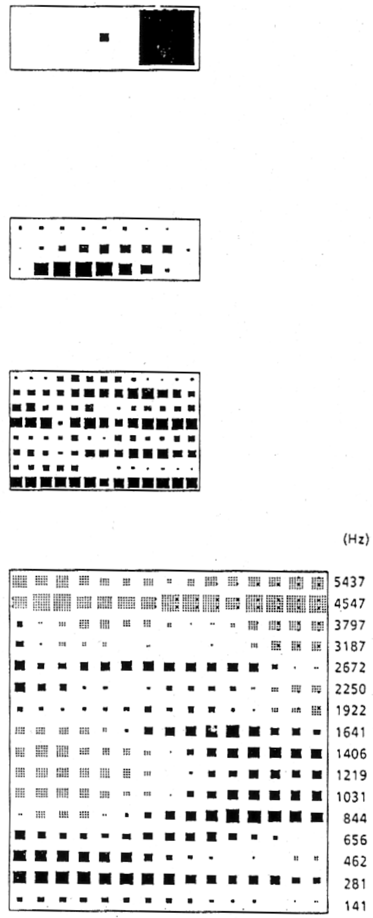
speaker	number of tokens	number of errors	recognition rate	TDNN	number of errors	recognition rate	HMM
MAU	b(227)	4	98.2	98.8	18	92.1	92.9
	d(179)	3	98.3		6	96.7	
	g(252)	1	99.6		23	90.9	
MHT	b(208)	2	99.0	99.1	8	96.2	97.2
	d(170)	0	100		3	98.2	
	g(254)	4	98.4		7	97.2	
MNM	b(216)	11	94.9	97.5	27	87.5	90.9
	d(178)	1	99.4		13	92.7	
	g(256)	4	98.4		19	92.6	

Caution:

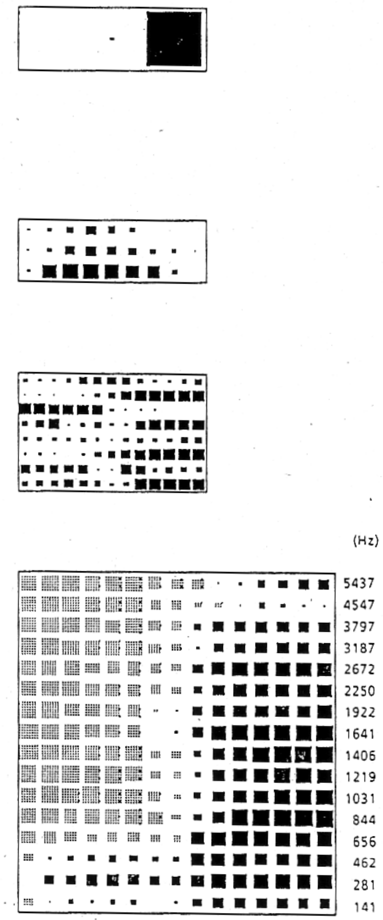
- HMM - Standard Model;
  more advanced models have been reported
- Different Front End Signal Processing



- Learned Acoustic-Phonetic Features:
  Formant Transitions, Segment Boundaries



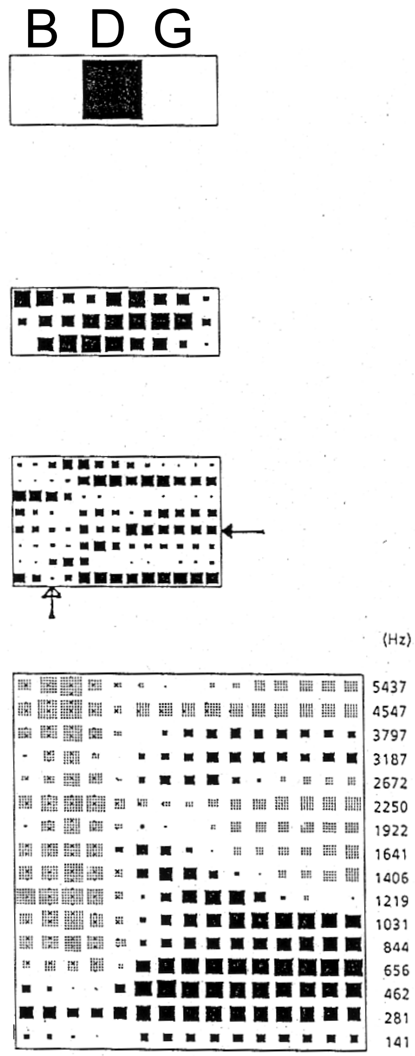
"GA" (word middle)



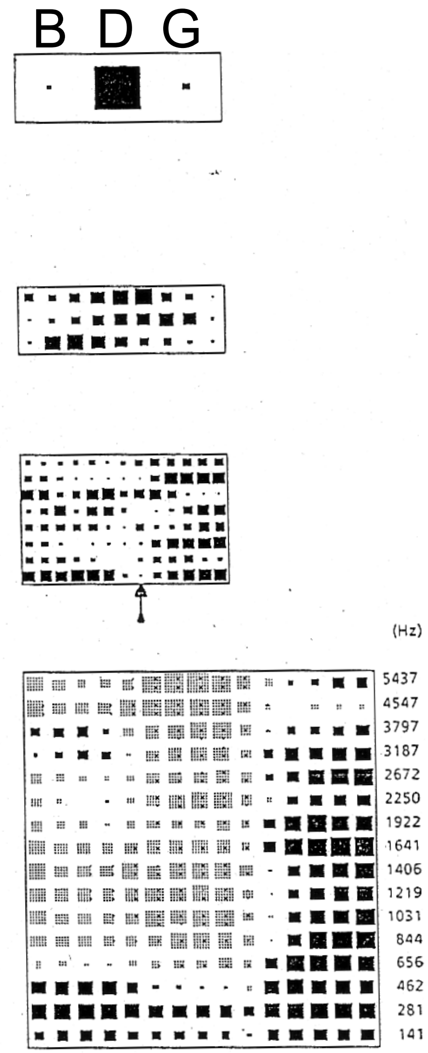
"GA" (word initial)

- Learned Alternate Internal Representations Link Different Acoustic Realizations to the Same Concept (Trading Relations)

KIT
Karlsruher Institut für Technologie



“DO” <- 30 msec



30 msec -> “DO”

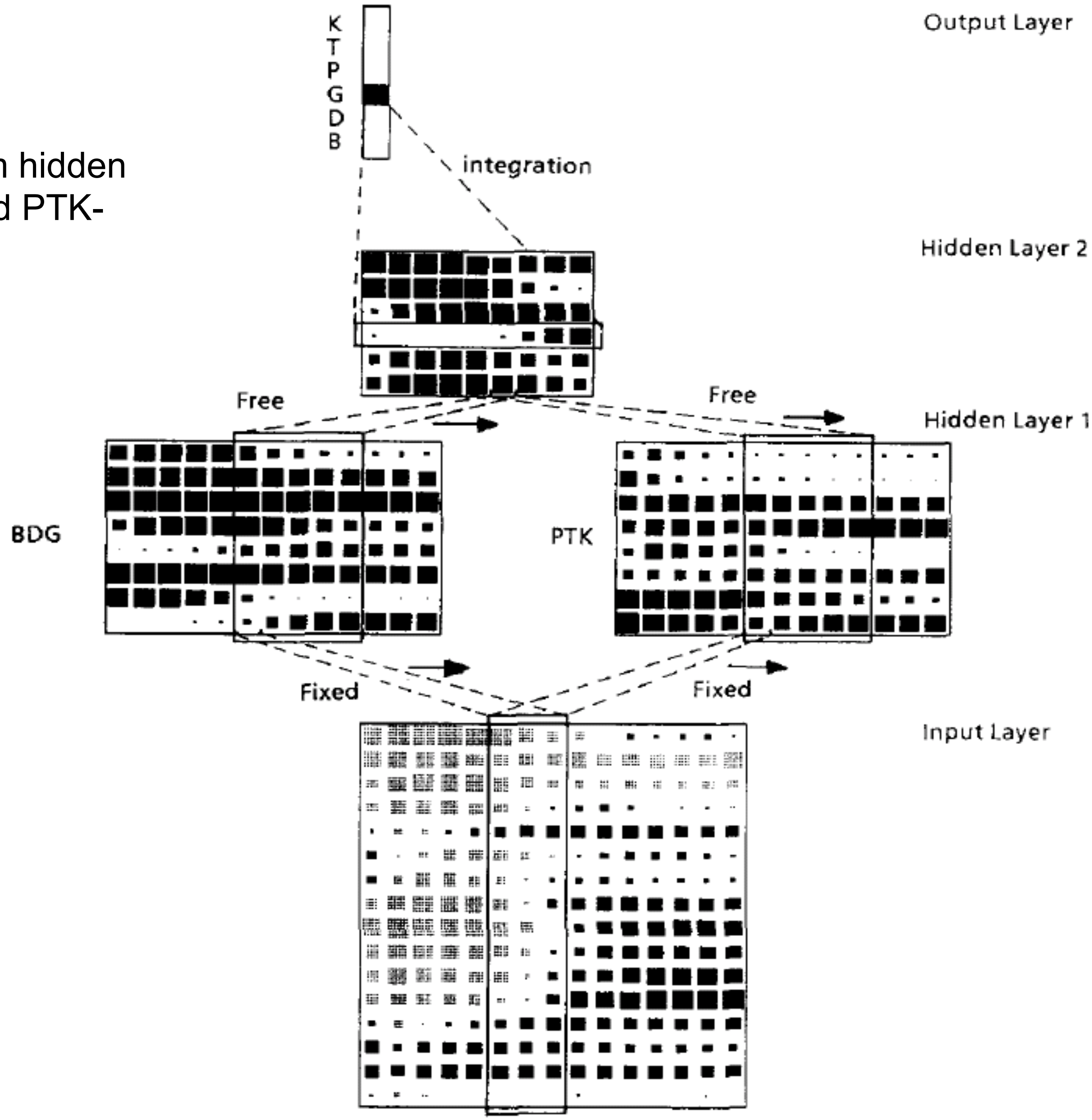
- Shift Invariance (in time)

Consonant Recognition Performance Results

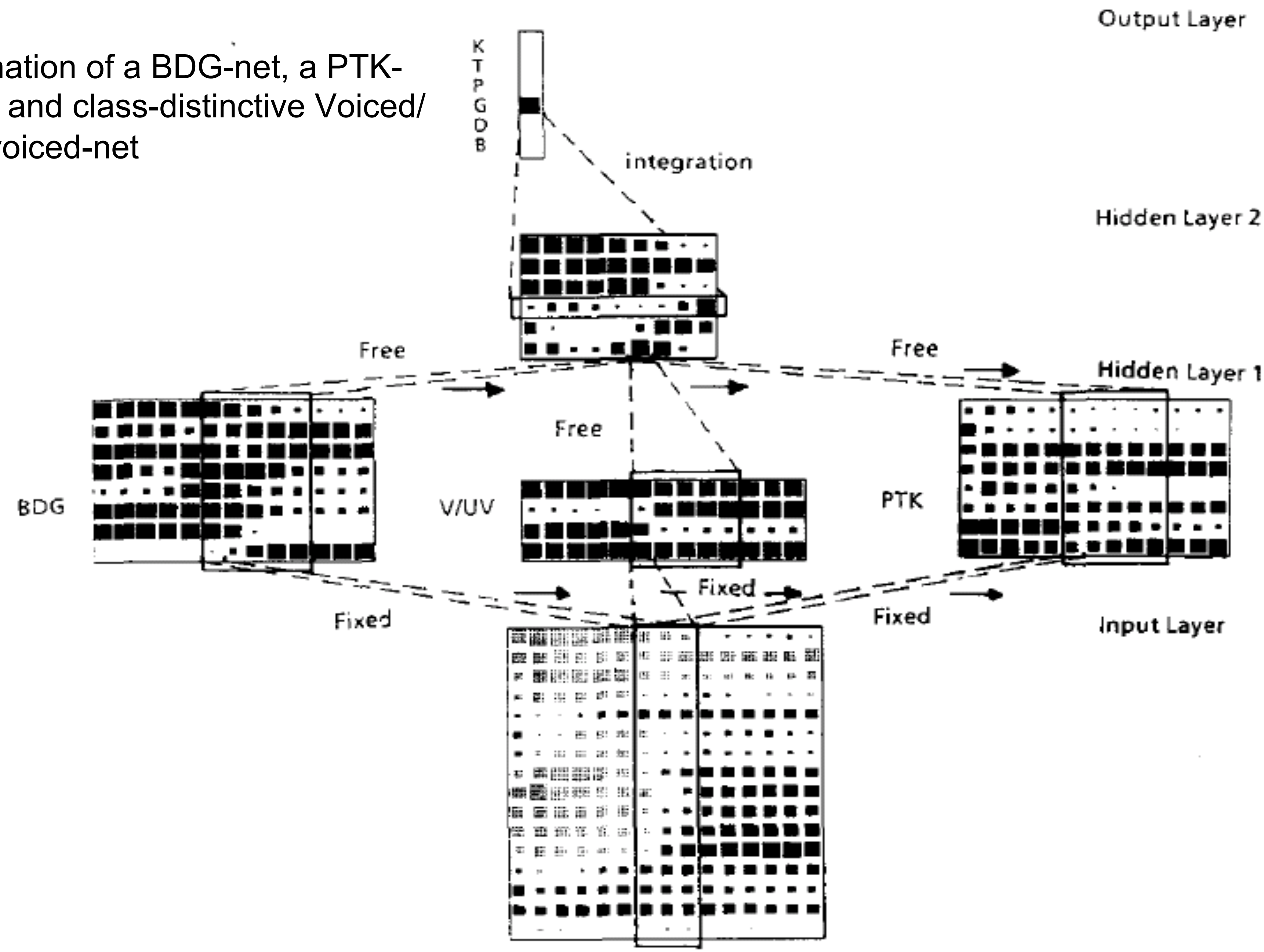
Task	Recognition Rate (%)
bdg	98.6
ptk	98.7
mnN	96.6
sshhz	99.3
chts	100.0
rwy	99.9
cons. class	96.7
All consonant TDNN	95.0
All-Net Fine Tuning	95.9
HMM(standard)	83.6
HMM(improved)	92.7

Karlsruher Institut für Technologie

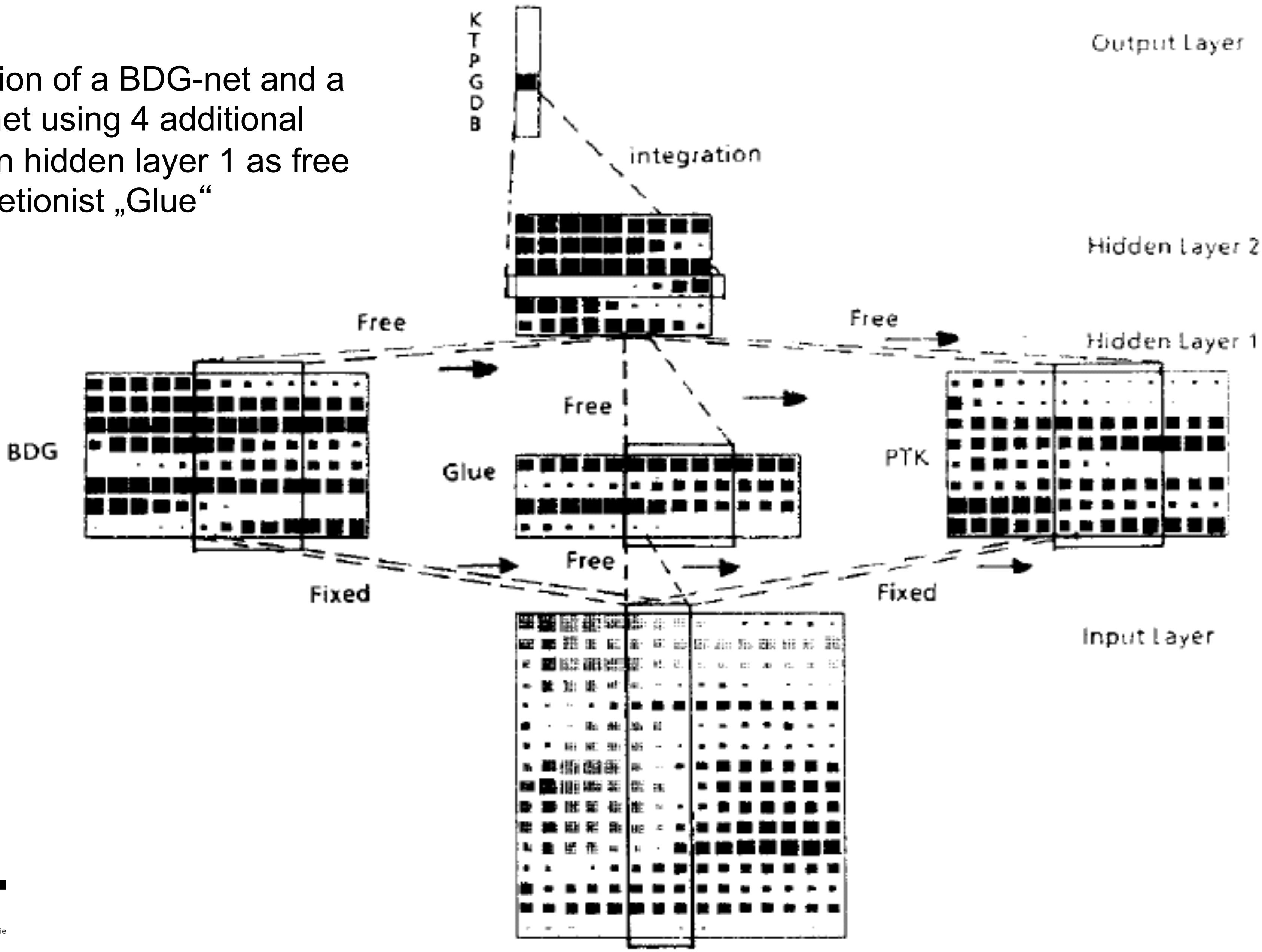
BDGPTK-net trained from hidden units from a BDG- and PTK-net



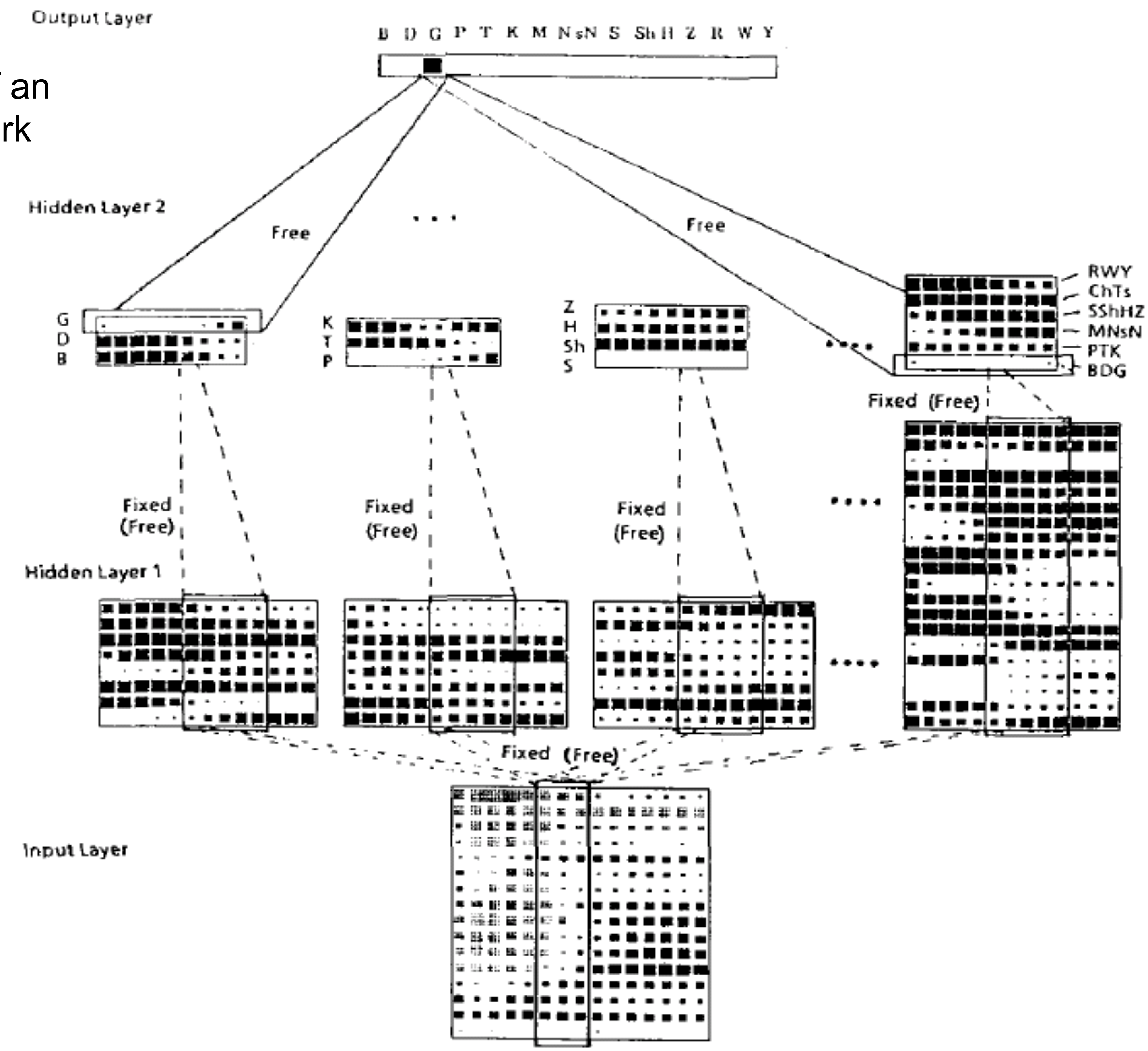
Combination of a BDG-net, a PTK-net, and class-distinctive Voiced/ Unvoiced-net



Combination of a BDG-net and a PTK-net using 4 additional units in hidden layer 1 as free conncetionist „Glue“



Modular construction of an all consonant network



Karlsruher Institut für Technologie

From BDG to BDGPTK: Modular Scaling Methods

Method	bdg	ptk	bdgptk
Individual TDNNs	98.3 %	98.7 %	
TDNN:Max. Activation			60.5 %
Retrain BDGPTK			98.3 %
Retrain Combined Higher Layers			98.1 %
Retrain with V/UV-units			98.4 %
Retrain with Glue			98.4 %
All-Net Fine Tuning			98.6 %

Consonant Recognition Performance Results

Task	Recognition Rate (%)
bdg	98.6
ptk	98.7
mnN	96.6
sshhz	99.3
chts	100.0
rwy	99.9
cons. class	96.7
All consonant TDNN	95.0
All-Net Fine Tuning	95.9
HMM(standard)	83.6
HMM(improved)	92.7

Word Models

- **Full word Templates:**

  Perceptron, Neural Net applied to input coefficient matrix

- **Problems:**
  - Time Alignment
  - Endpoint Detection
  - Large Vocabularies (Training Data, Time)

Word Models (cont.)

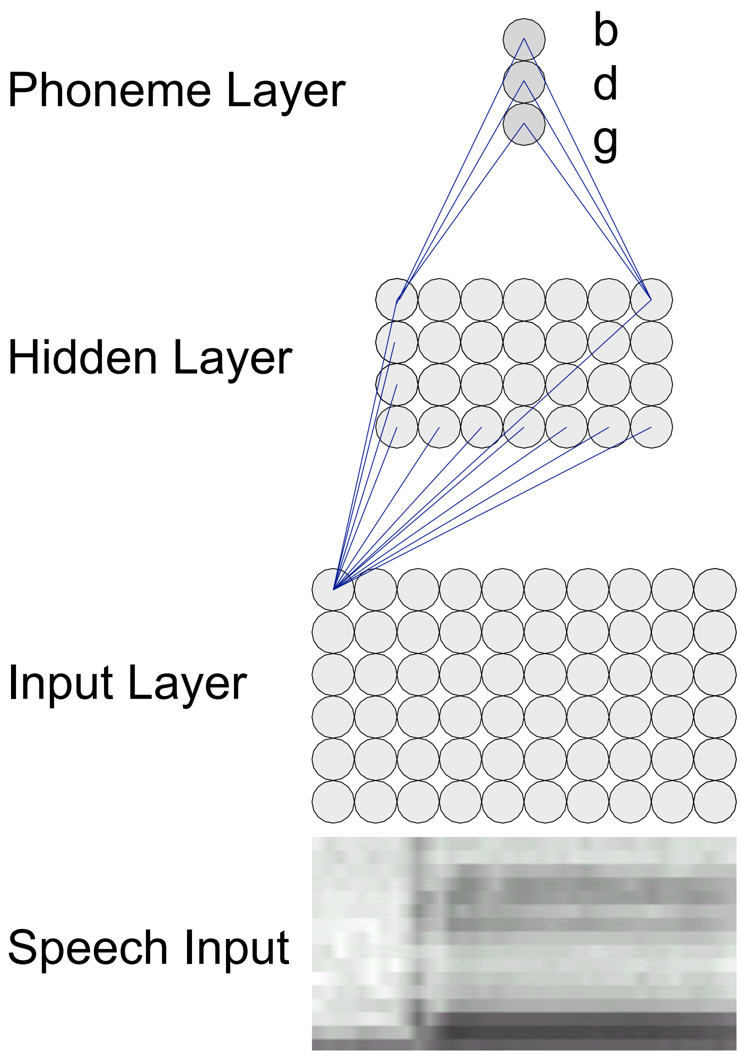
Time Alignment, Endpoint Detection

- Dynamic Neural Net (Sakoe)
- Word Level TDNN (Bottou)
- Time Delay (Tank & Hopfield)
- Preprocessing Time Alignment (Burr)
- Neural Prediction Model (Iso)
- Hidden control Neural Network (Levin)

Large Vocabularies:

- Model/Classify Atomic Subword Units (Phonemes, Phones, States)
- Integrate while Optimizing for Word Recognition

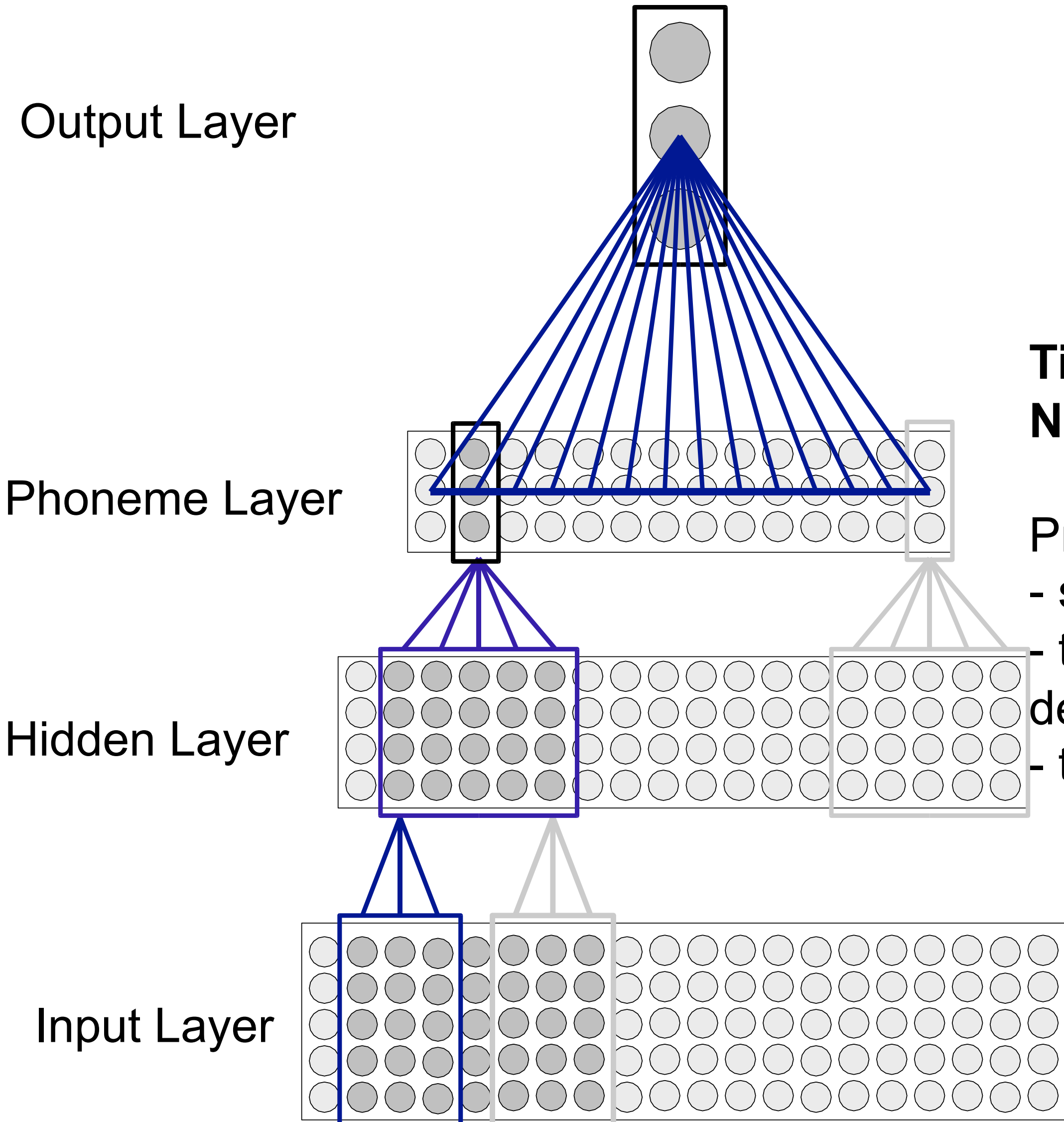
Multi-Layer Perceptron



Phoneme Classification (b,d,g) with a **fully connected** MLP

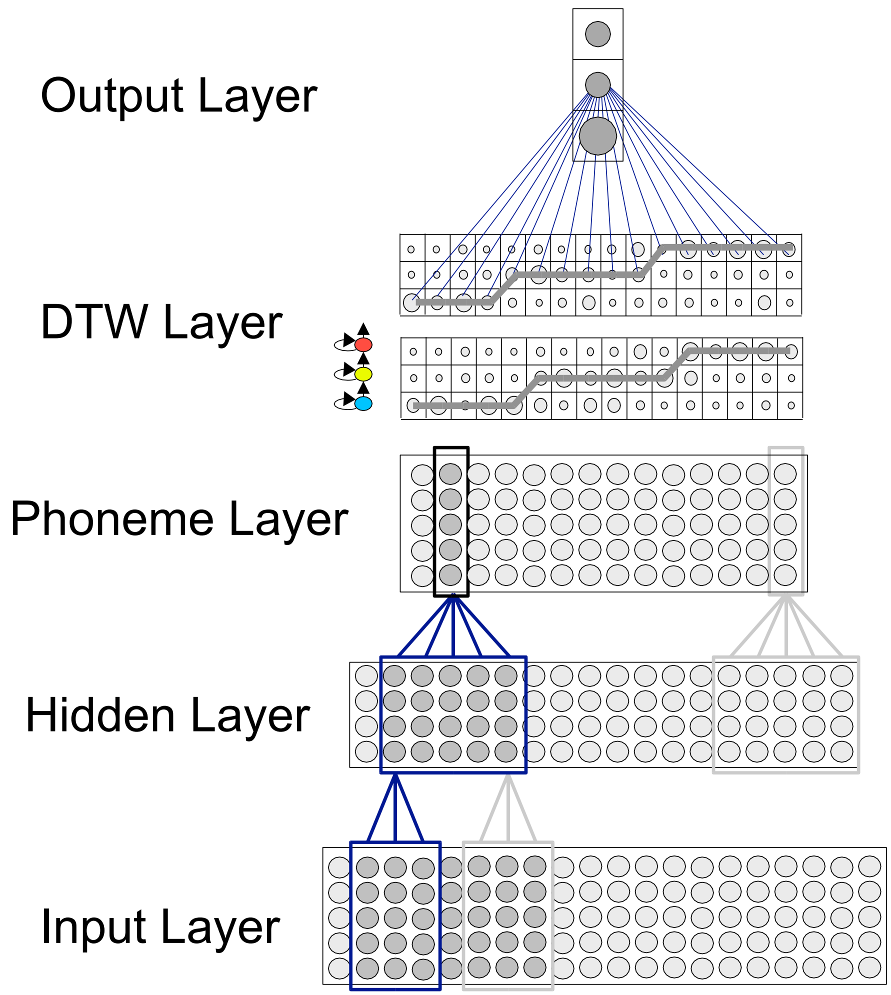
Problem:
static network,
dynamic input

Time Delay Neural Network

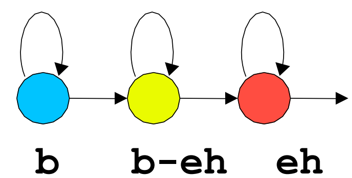


Time Delay Neural Network (TDNN)

Properties:
- shift invariant
- temporal context (time delays)
- temporal integration

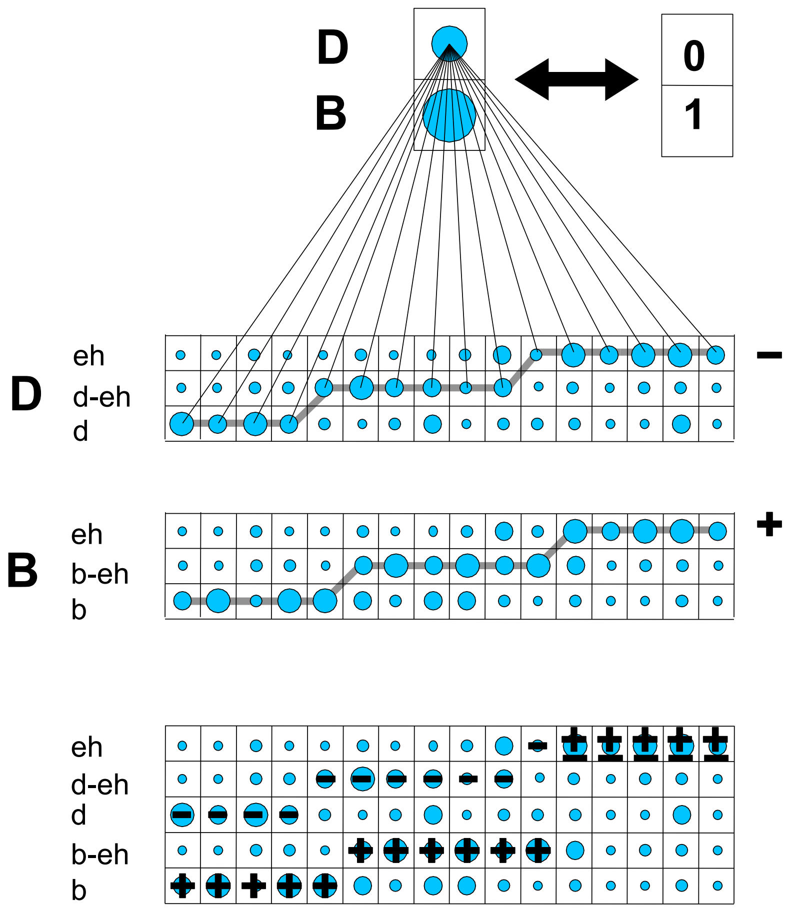


Letter 'B':



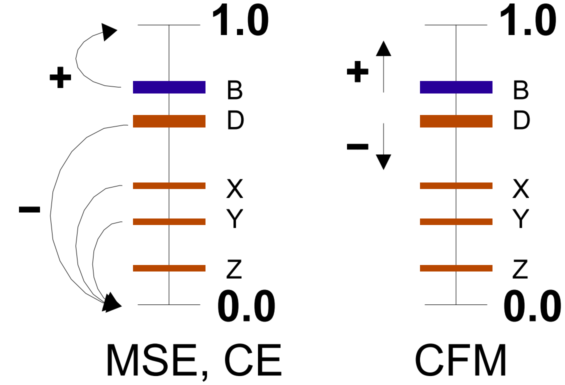
=> Multi-State TDNN

Training on Letter Level



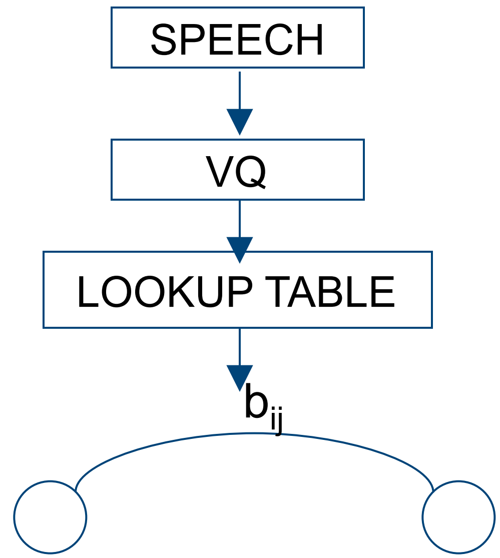
Error Function

- mean square error (MSE)
- cross entropy (CE)
- classification figure of merit (CFM)

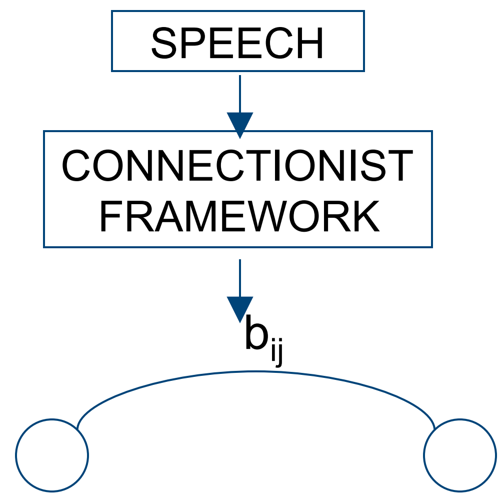


Fitting Connectionist into HMMs

Computing output probabilities
in a typical discrete HMM



Computing output probabilities in a hybrid connectionist/HMM system



Hybrid MLP/HMM

Motivations

- Discriminative training
- Combine multiple features without assuming independence
- Sharing and flexible allocation of representational resources
- Model correlations

MLPs can compute posterior class probabilities (Bourlard & Wellekens)

Use MLP to estimate HMM observation likelihoods in DECHIPHER

Initial context-independent integration:

$$P(Y_t \mid q_j) = \frac{P(q_j \mid Y_t) P(Y_t)}{P(q_j)}$$

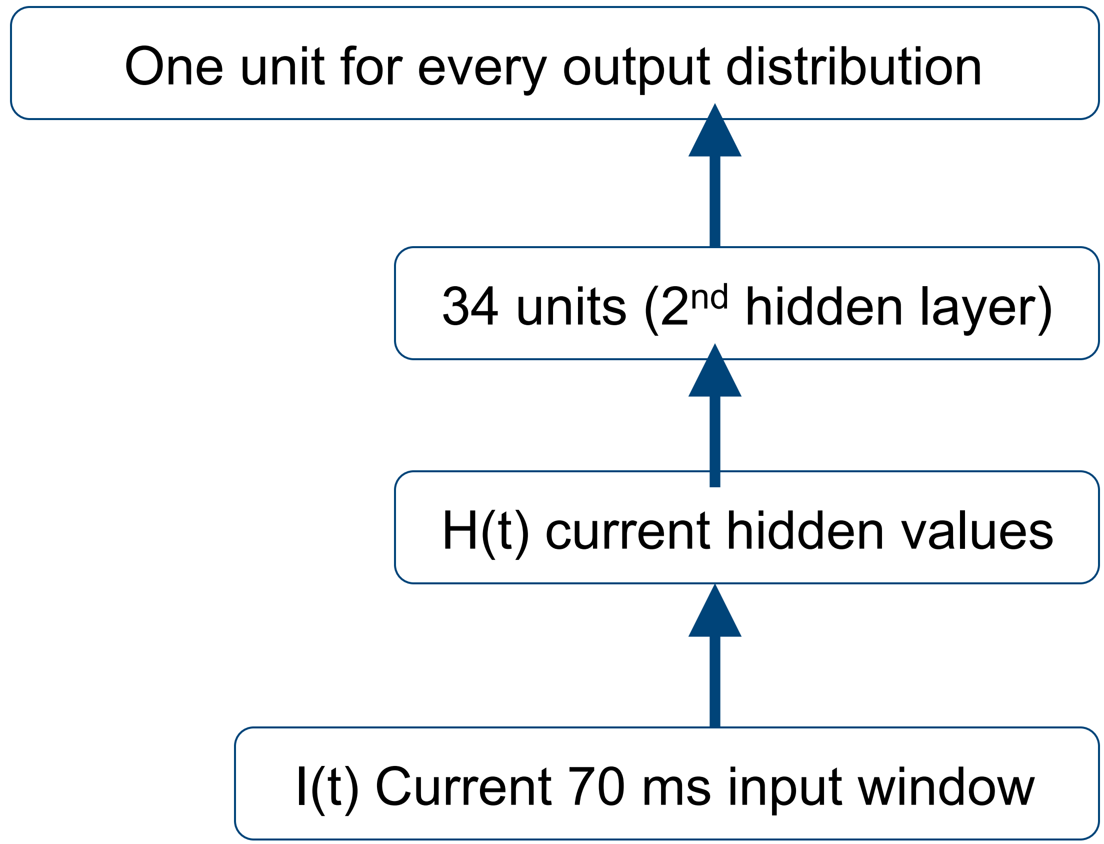
KIT Karlsruher Institut für Technologie

NN-HMM Hybrid Methods

- Idea:
  - Neural Net for classification of phones/states
  - HMM for alignment and integration into words
- Approach:
  - Output activations => Maximum a posteriori probabilities (Bourlard)
  - Log [Word Probability] => Σ log [Output Activations] along best alignment path
  - Alignment path is determined by DTW or Viterbi alignment

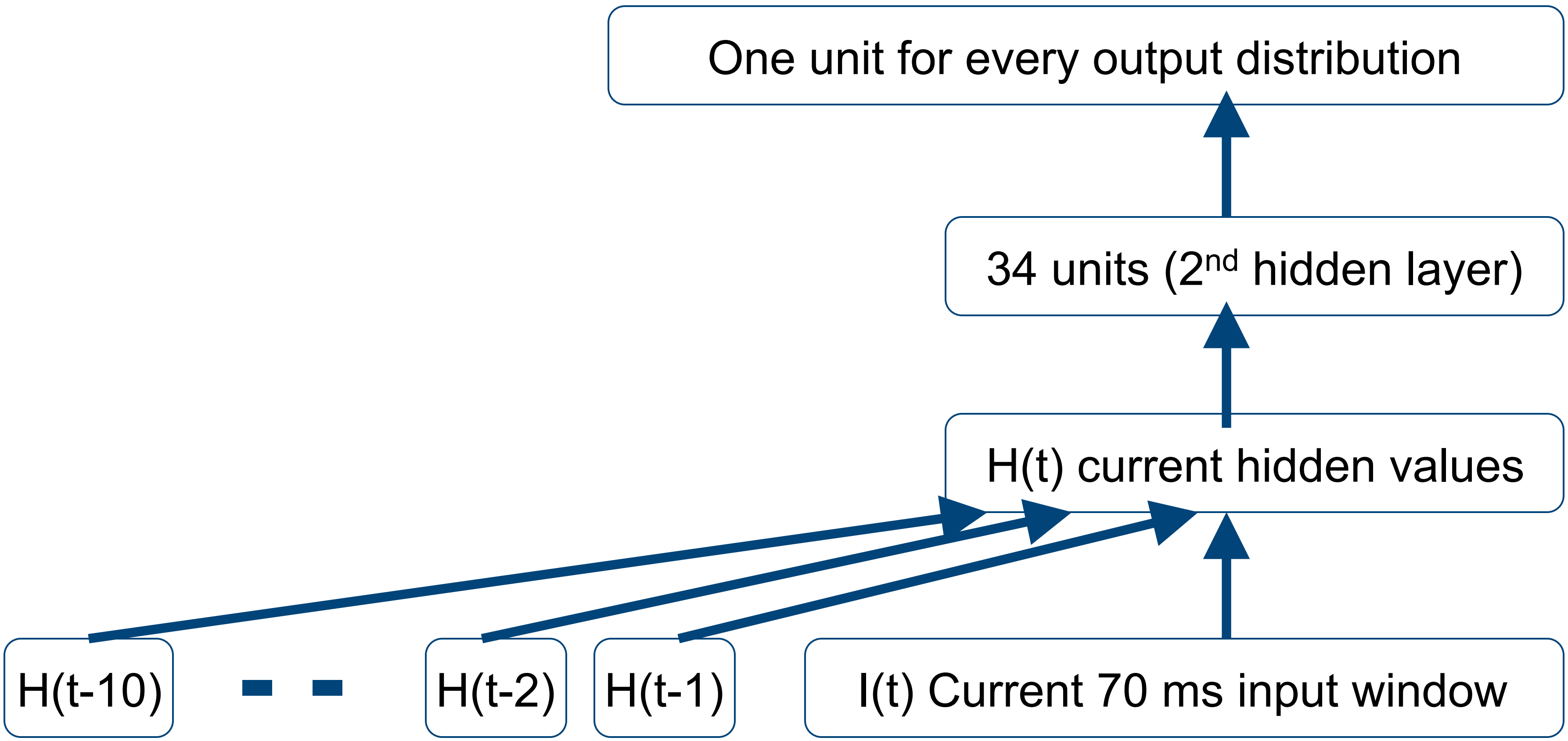
Karlsruher Institut für Technologie

Network Topology (1991)



The non-recurrent CVT network

Network Topology



The recurrent CVT network

Performance on TI Digit Task A Comparison of Various Systems

Doddington	1990	99.5%	98.6%
Franzini Waibel Lee	1991	99.1%	98.0%
	1990	98.5%	95.0%
	1989	97.0%	91.0%
Levin	1990	99.1%	
Rabiner	1990		97.0%
Sphinx	1988	97%	92%

Results Hybrids

	word acc.	string acc.	Δerror rate
Feb '90	98.5%	95.0%	
- recurrence	98.0%	94.7%	+6%
+ word models	98.7%	96.8%	-40%
+ corrective training	98.8%	97.0%	-6%
+ multiple models	99.1%	98.0%	-33%

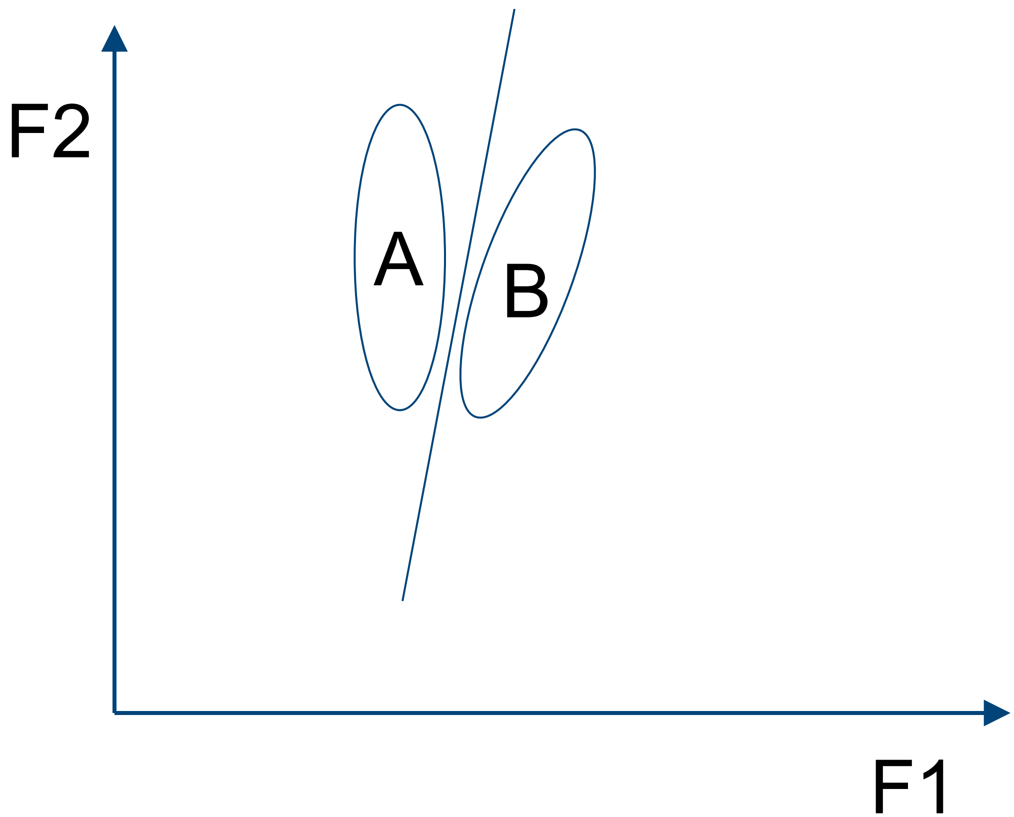
NN-HMM Hybrid Improvements

- Doddington, Bourlard, Wellekens
  - Viterbi Training; Iterative Alignment and Training
  - Normalized by Priors
  - Word Transition Penalties
  - Cross Validation Set
- Franzini
  - Connectionist Viterbi Training
  - Recurrence
  - Phone Modeling, Word Modeling
  - Word Corrective Training
  - Multiple Models

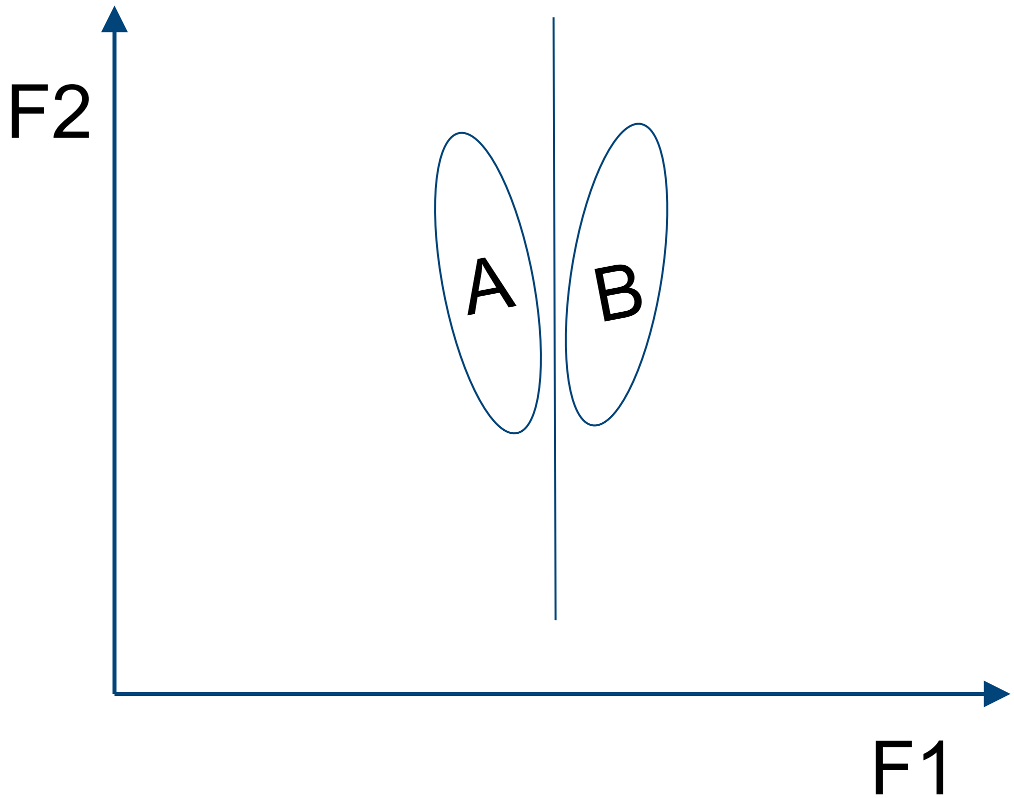
Speaker Independence

Neuronale Netze u. Anwendungen,
15. Juni 2004

Speaker 1



Speaker 2



Speaker and Environmental Variability

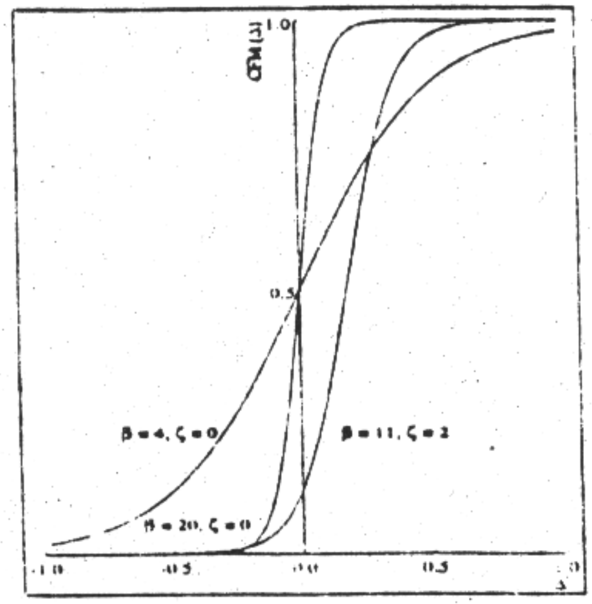
- Robust Objective Functions
  - Mean Square Error (MSE)
  - Classification Figure of Merit (CFM)
  - Cross Entropy

- Model Invariance
  - Frequency Shift Invariance
  - Invariance towards Tilt, Compression, etc.

Speaker and Environmental Variability (cont.)

- Adaptation
  - Human Perception (1,2 Syllables)
  - Slow Adaptation - Modify Weights
  - Fast Adaptation - Select (Mix of) Pretrained Specific Submodels

- Normalization
  - Environment- Correcting for Signal Noise
  - Speaker- Mapping New Speaker to Standard Speaker



CFM plotted for representative parameter values

Correct Output is D:

O(D)-O(B) O(D)-O(G)

O B

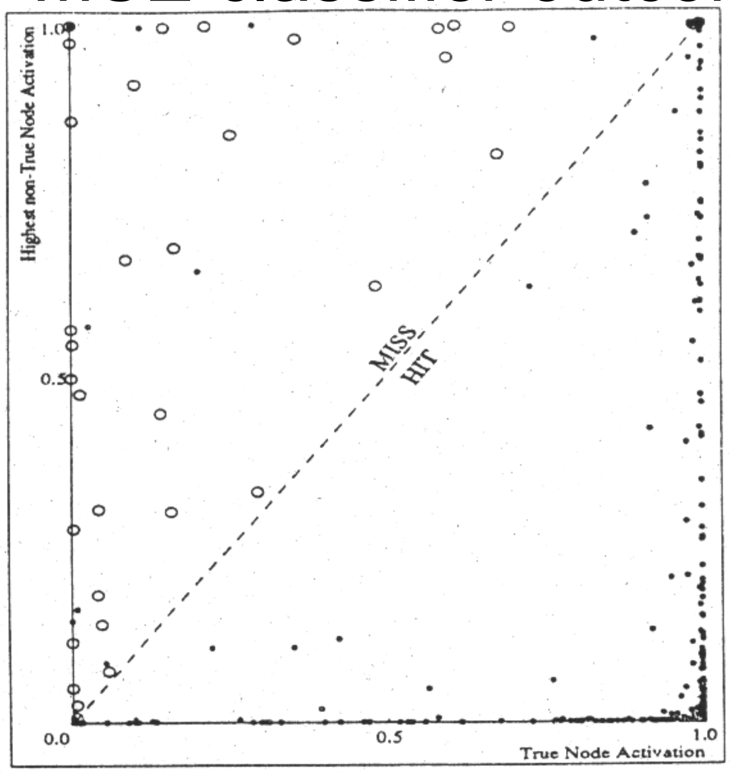
O D

O G

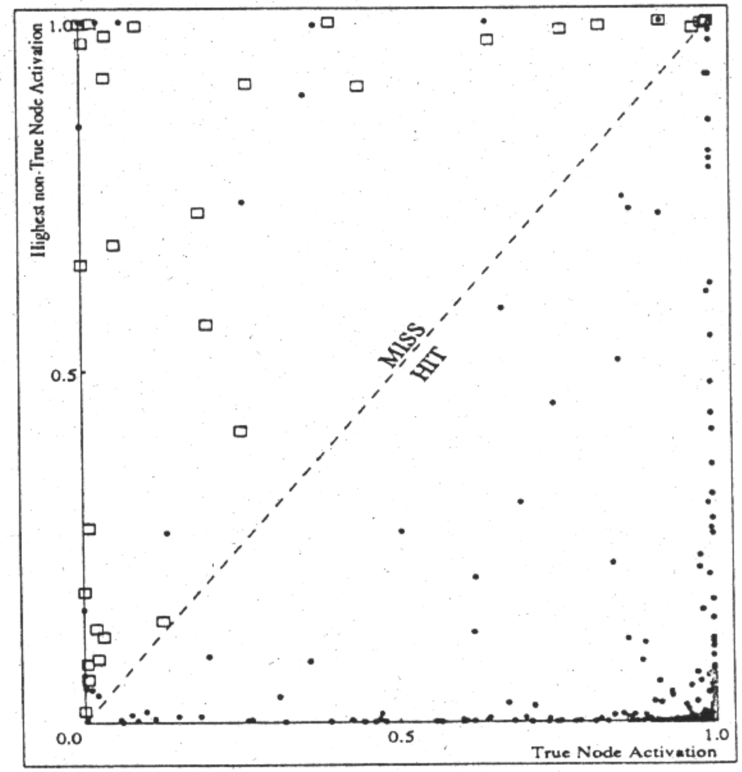
Output units

Scatter plots of MSE and CFM classifier

MSE classifier outcome



CFM classifier outcome



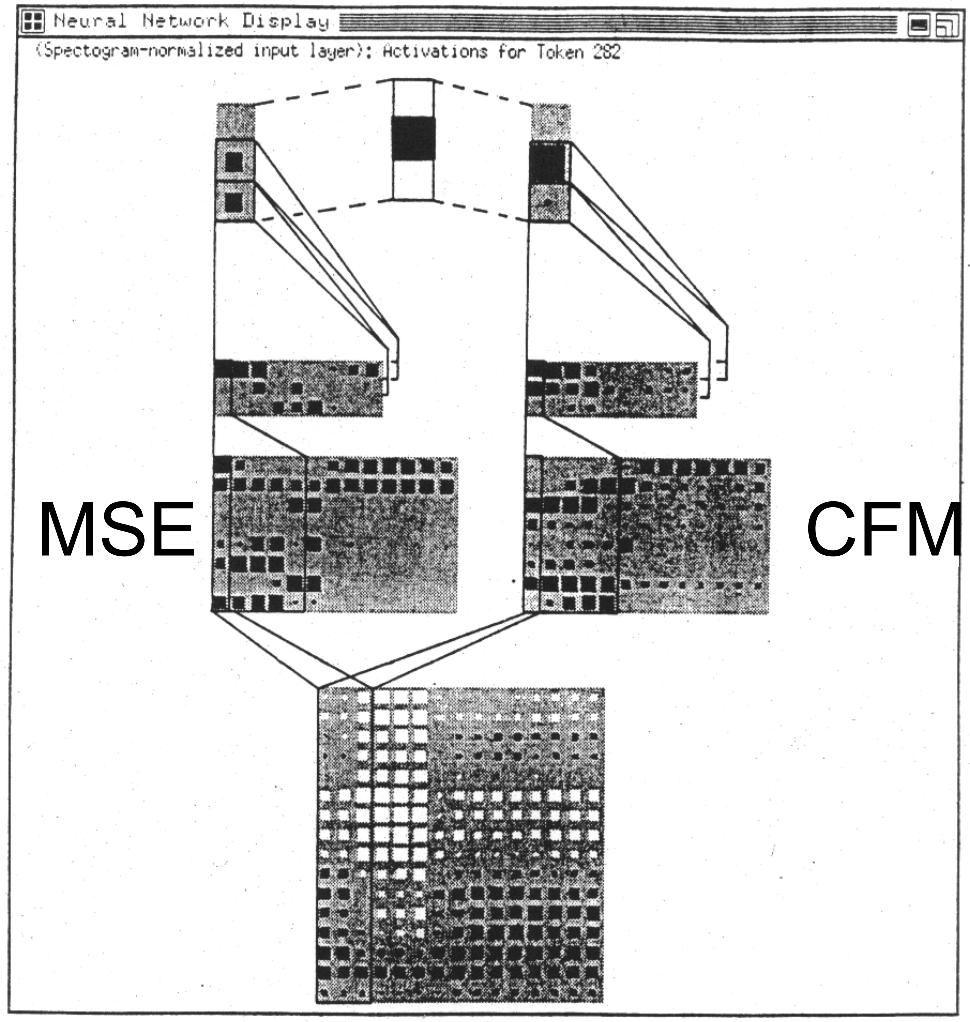
- ○ Indicates MSE miss correctly classified by CFM
- □ Indicates CFM miss correctly classified by MSE

Two standard TDNNs

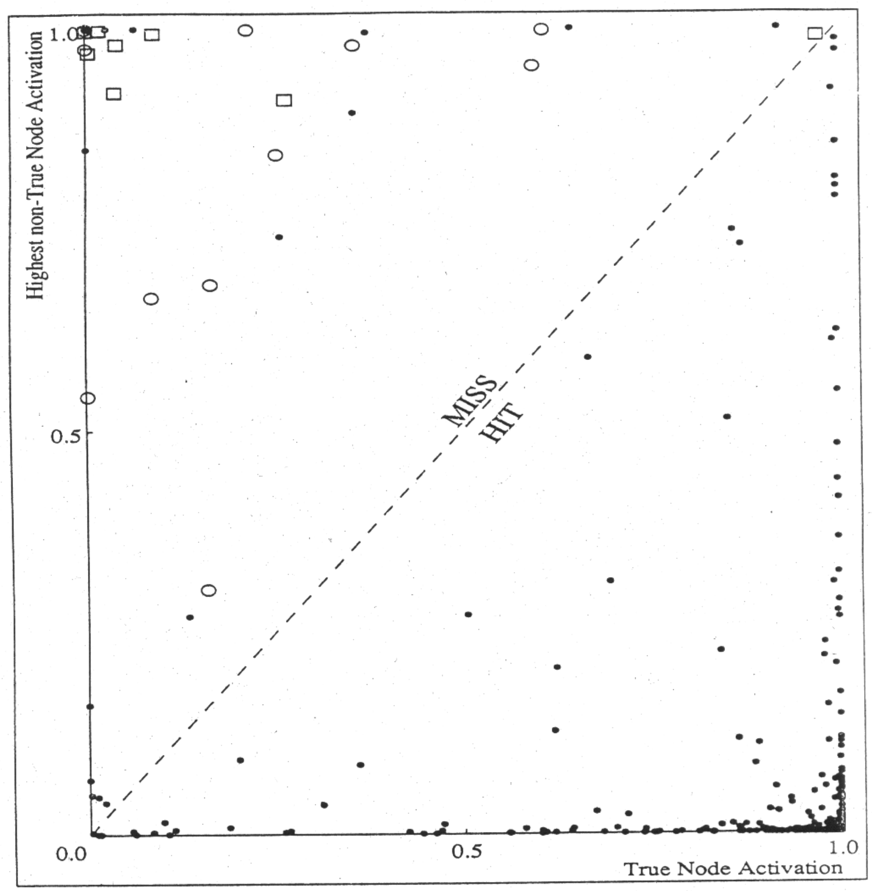
One trained with MSE objective function, the other with the CMF objective function.

The MSE-trained network yields an ambiguous classification, but the CFM-trained network yields a confident, unambiguous classification.

Through a simple arbitration scheme, the combined classifiers yield the correct classification.



Scatter plot of arbitrated MSE/CFM classifier outcomes



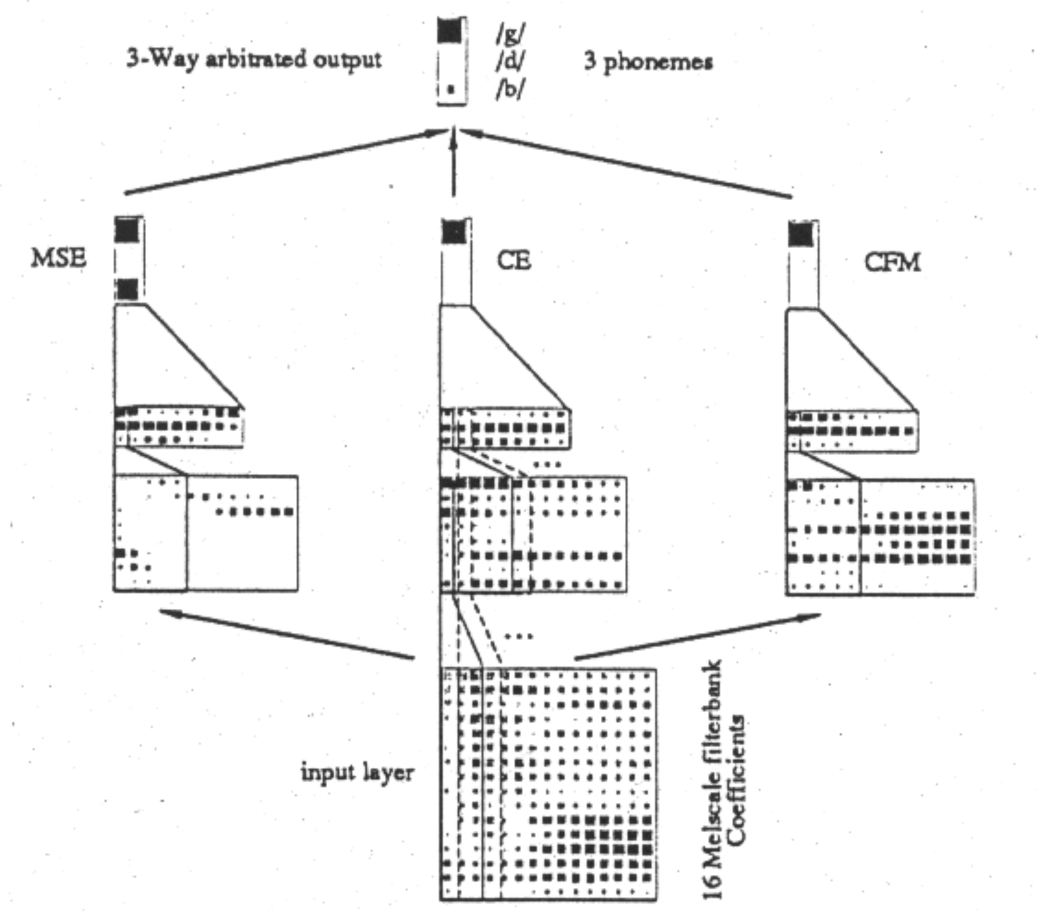
- □ Indicates post arbitration miss correctly classified by MSE
- ○ Indicates post arbitration miss correctly classified by CFM

Comparison of /b, d, g/ recognition rates for TDNN

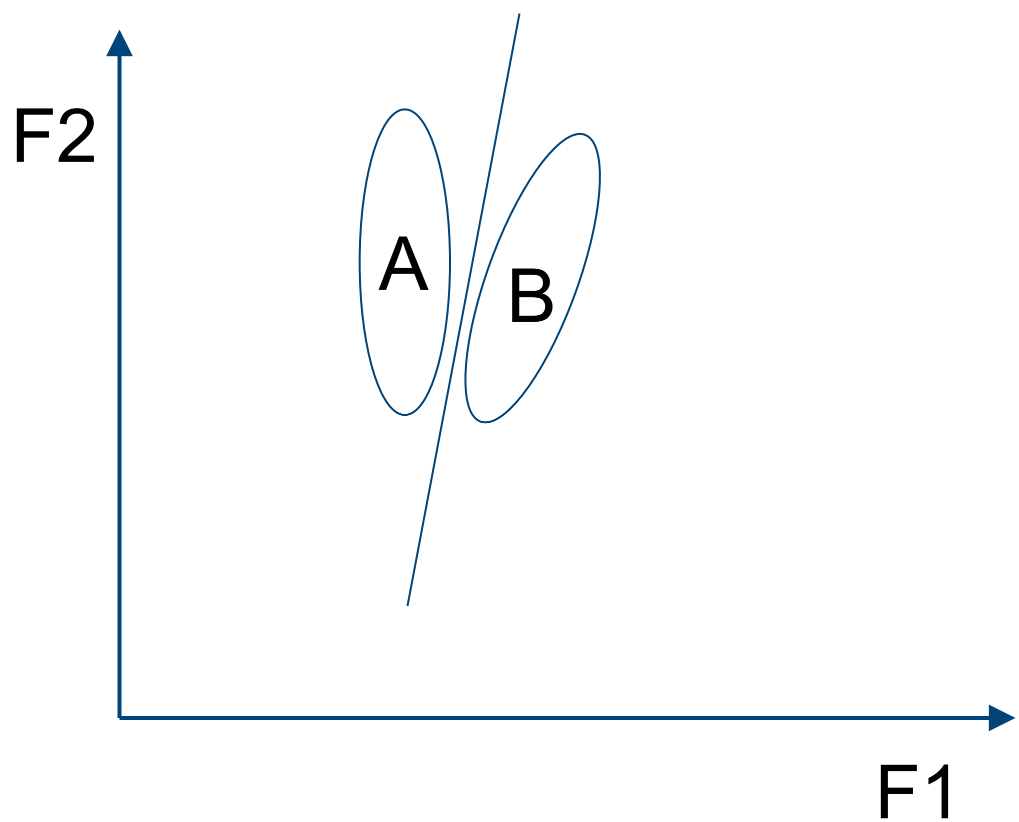
Network	Speaker	MSE	CFM	MSE/CFM	Mod MSE/CFM
TDNN	MAU	98.3	98.9	98.8	98.8
	MHT	99.7	99.5	99.7	99.7
	MNM	97.4	97.2	98.3	97.8
	FKN	97.6	97.8	98.1	98.3
	FSU	98.2	98.5	98.4	98.4
	MMS	97.7	98.5	98.5	98.6
TDNN	1 st 3	97.3	97.5	98.1	98.3
	all 6	95.9	95.9	96.5	96.5

Trained with MSE, CFM and arbitrated MSE/CFM objective function
(CFM parameters: $\alpha = 1.0$, $\beta = 4.0$, $\zeta = 0.0$)

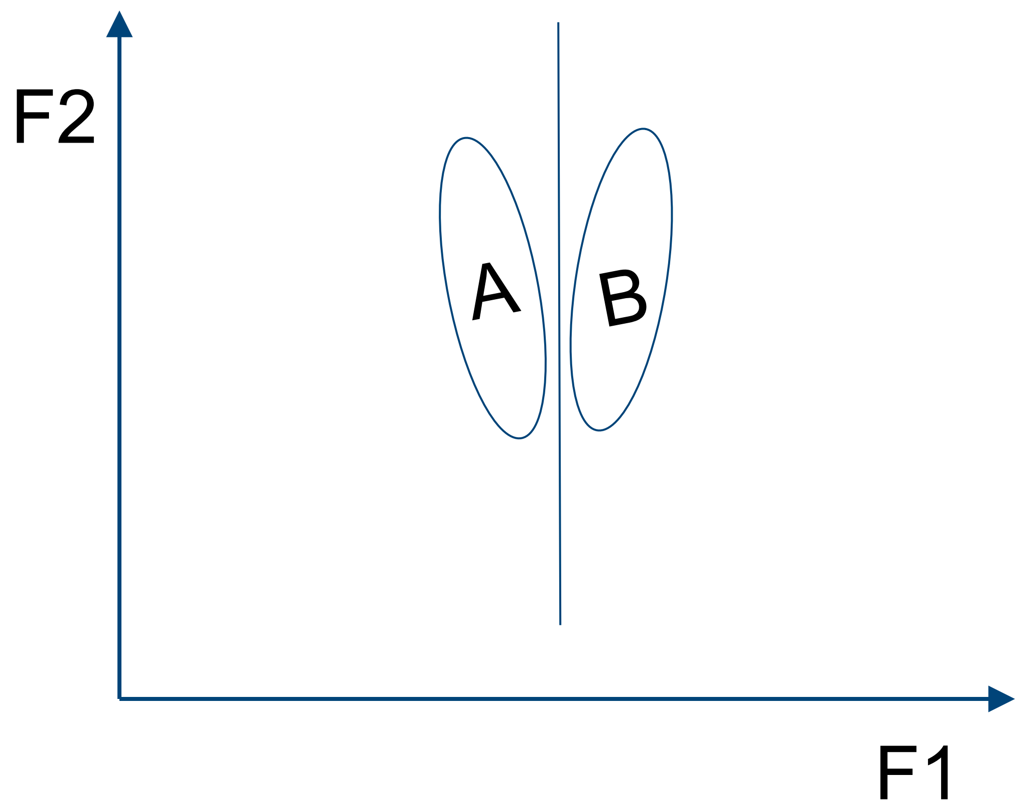
TDNN 3-Way arbitrated output



Speaker 1

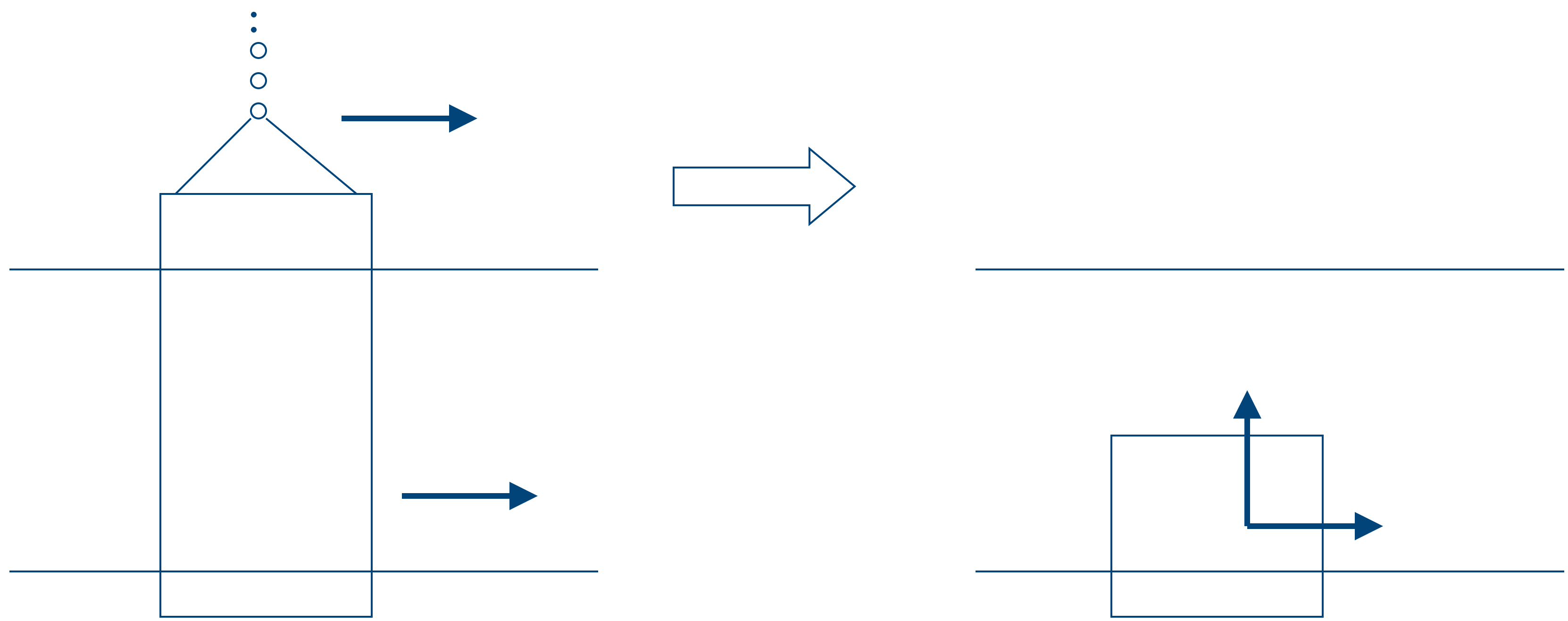


Speaker 2

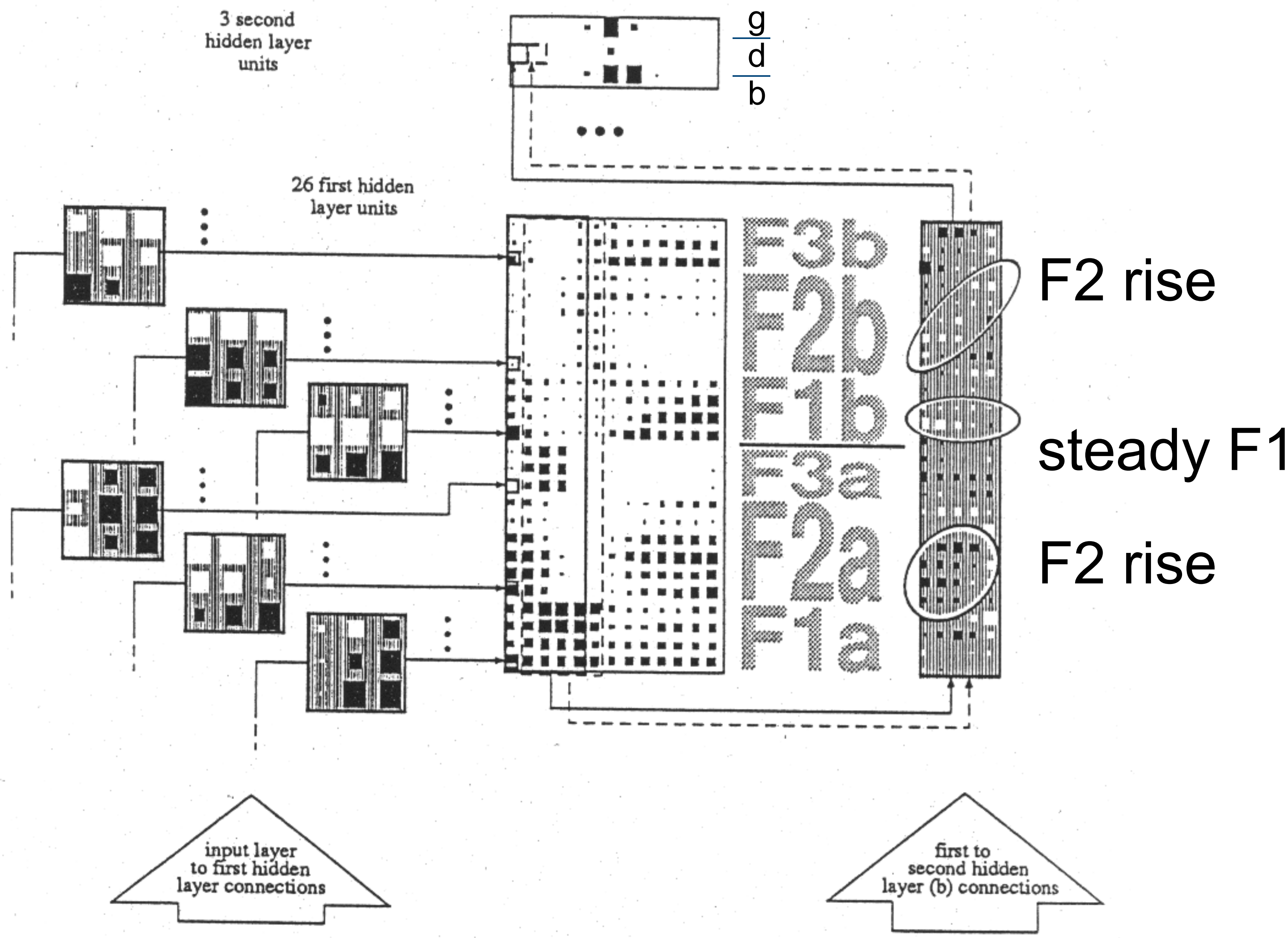


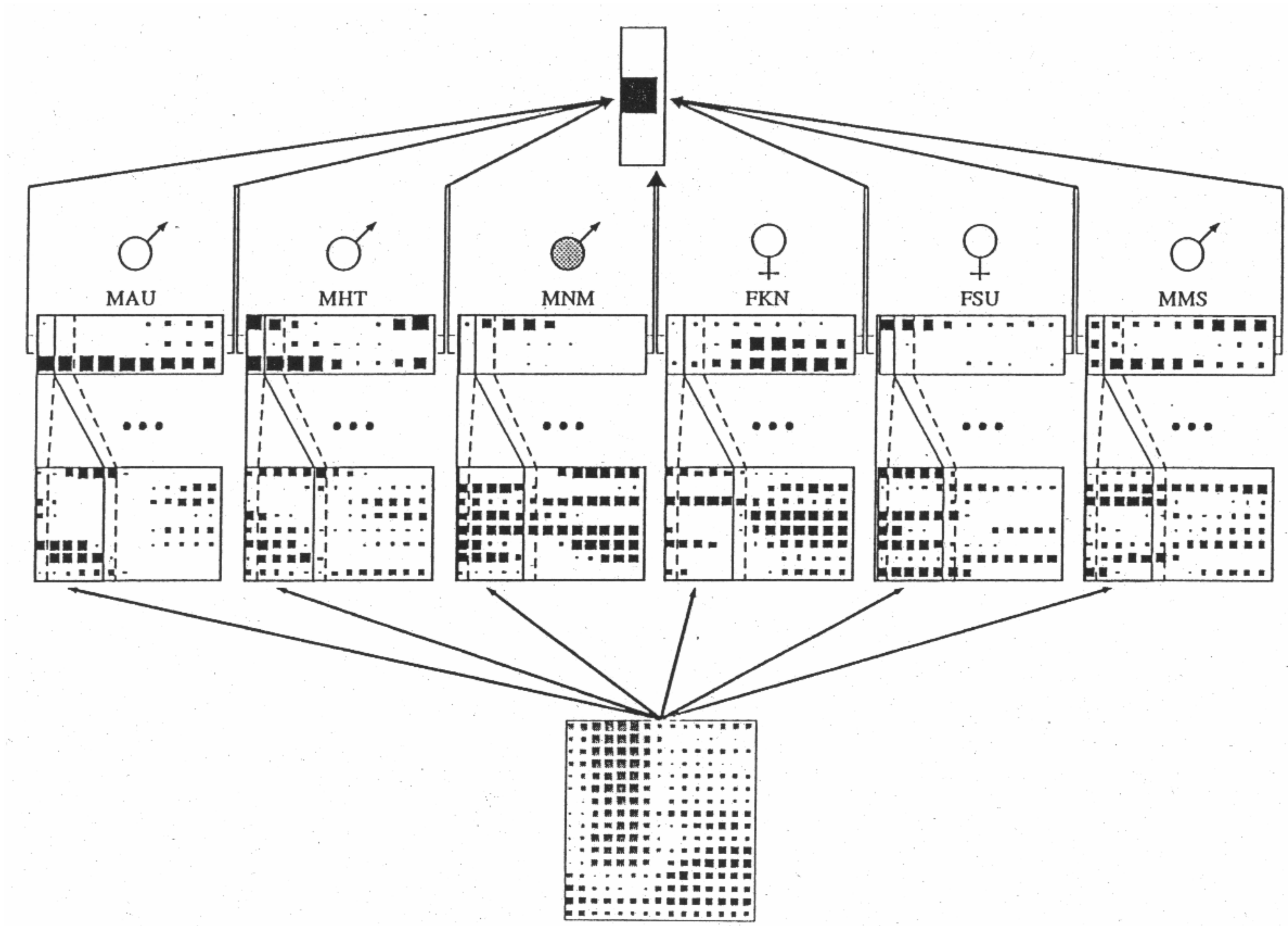
Speaker Independence

Shift Invariance in Frequency



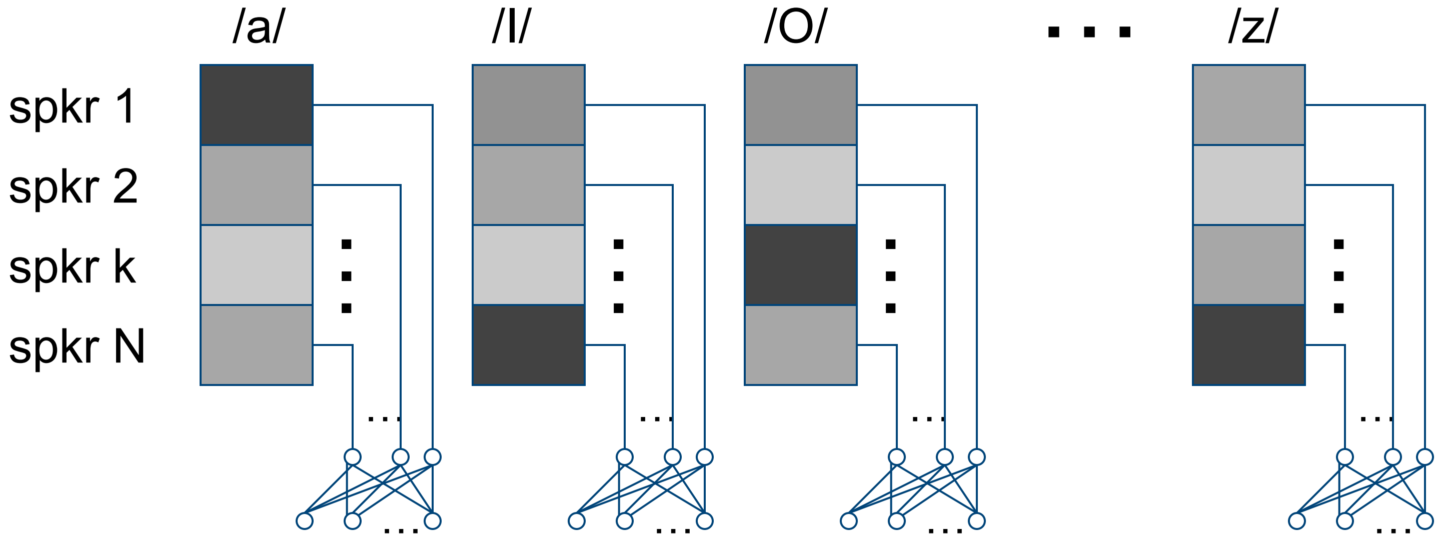
So far, no improvements in performance


3 second
hidden layer
units
g
d
b
26 first hidden
layer units
F3b
F2b
F1b
F3a
F2a
F1a
F2 rise
steady F1
F2 rise
input layer
to first hidden
layer connections
first to
second hidden
layer (b) connections

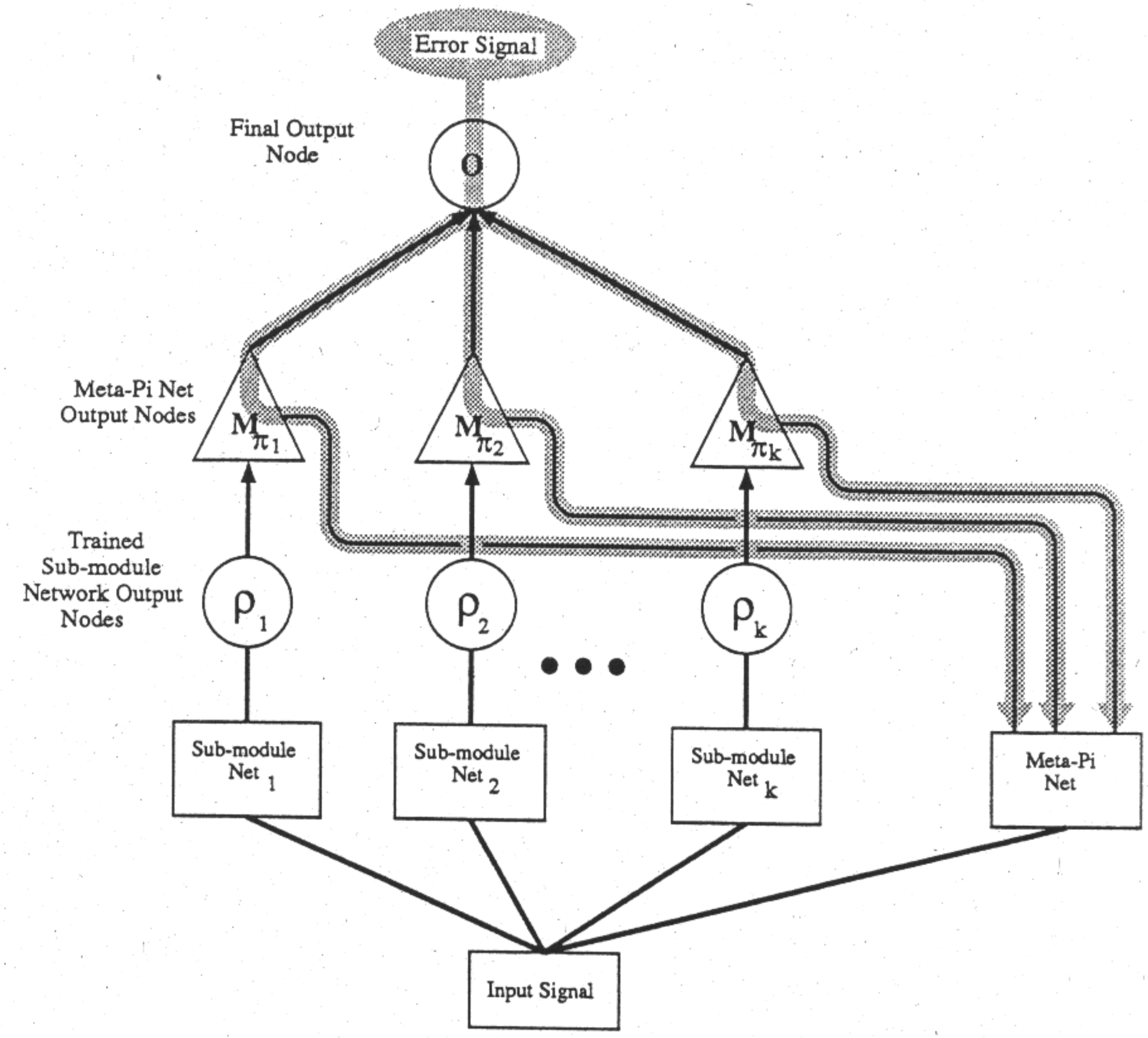

MAU
MHT
MNM
FKN
FSU
MMS

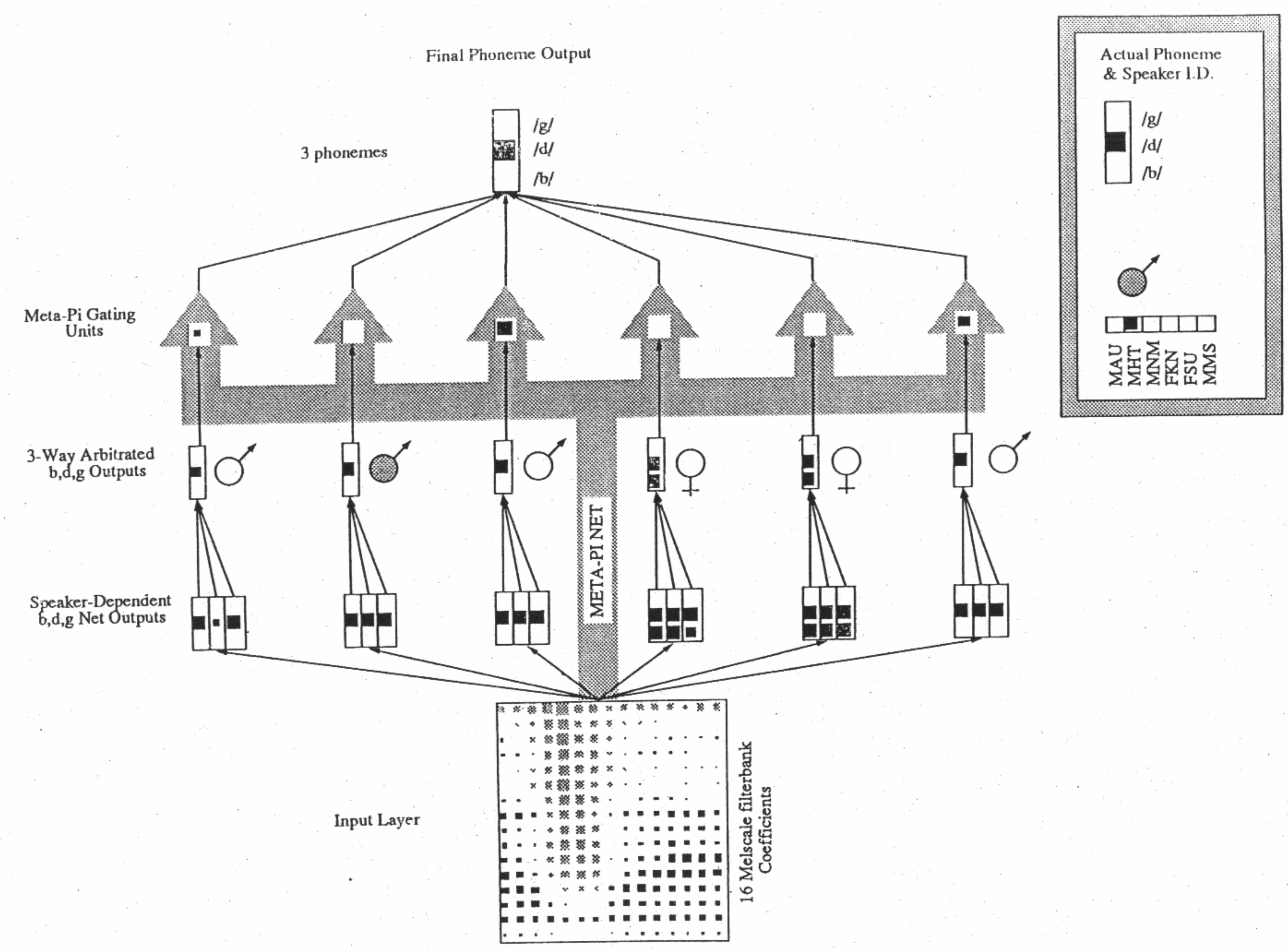
Multi-Speaker Reference Model

A speaker-specific reference model is composed from several well trained reference models



phoneme specific
reference model selection networks


Error Signal
Final Output Node
o
Meta-Pi Net Output Nodes
Trained Sub-module Network Output Nodes
Sub-module Net 1
Sub-module Net 2
Sub-module Net k
Meta-Pi Net
Input Signal



Final Phoneme Output
3 phonemes
/g/
/d/
/b/
Actual Phoneme
& Speaker I.D.
/g/
/d/
/b/
MAU
MHT
MNM
FKN
FSU
MMS
Meta-Pi Gating Units
3-Way Arbitrated b,d,g Outputs
META-PI NET
Speaker-Dependent b,d,g Net Outputs
Input Layer
16 Melscale filterbank Coefficients

Multi-Speaker Results

1. Average performance of 6 speaker dependent nets

   98.7%

2. Performance of multi-speaker TDNN; trained on all 6 speakers, evaluated on different test data, but one of the speakers:

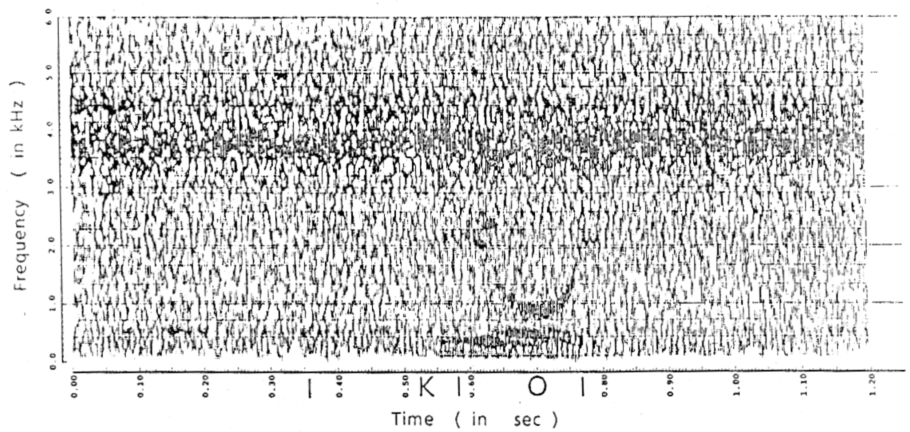
   95.9%

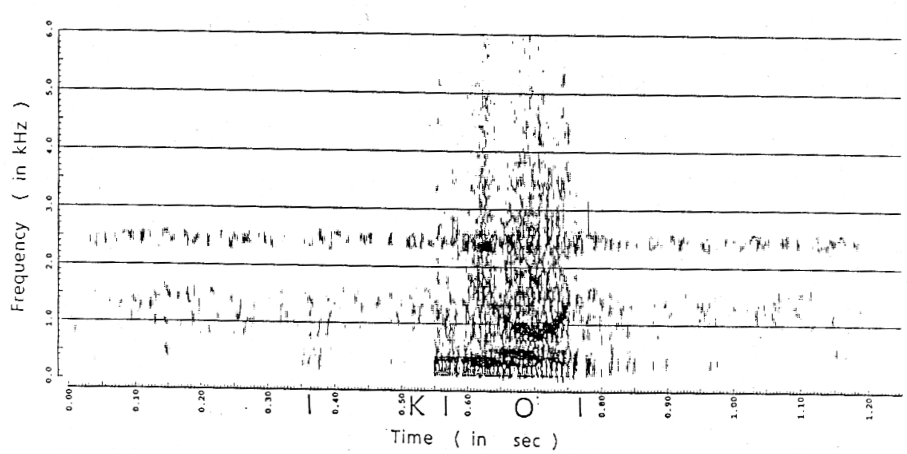
3. Meta-Pi Net:

   98.4%

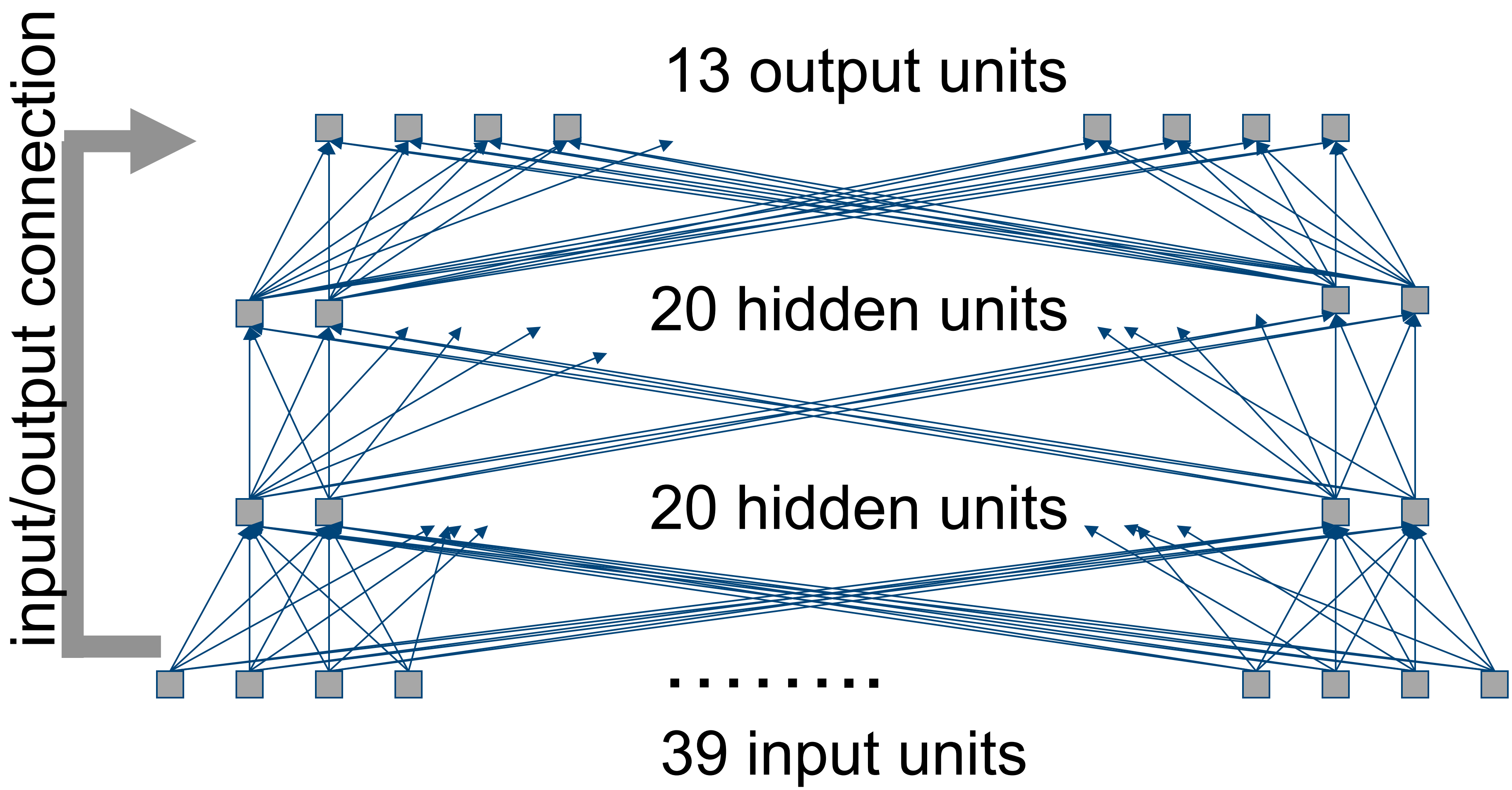
Spectrograms of the Network Input and Output -Example 1

(Speaker: Training, Word: Training, Noise: Training)





Speaker Net


input/output connection
13 output units
20 hidden units
20 hidden units
39 input units

Speaker Normalization Results

- Speaker-dependent models (2400)
- The error rate for the other speakers is 41.9%
- With 40 text-dependent training sentences, the error rate is reduced to 6.8%

Speakers	Without Norm	With Norm
JLS	8.5%	6.8%
BJW	62.1%	4.2%
JRM	55.3%	9.5%
Average	41.9%	6.8%

Speaker normalization error rates

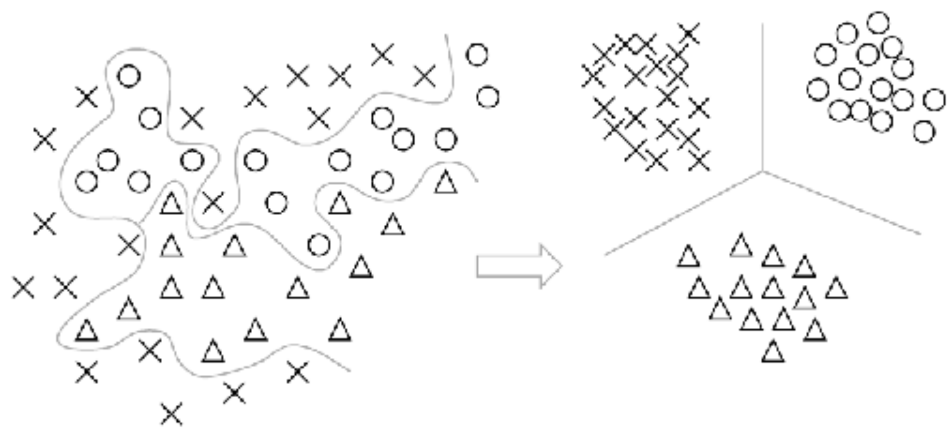
Feature Extraction in ASR

- Input: real world audio signal
- Goal: output a sequence of vectors that contain only the most *useful* information
- Problems:
  - What is useful information?
  - What vector rate should we choose?
  - What size should the feature vectors be?
- For detailed answers see ASR lecture
- Some techniques lead to large vectors
  - stacking, multi-resolutions, Δ, $\Delta\Delta$, ...
- => Problematic for the ASR system
  - speed, resource requirements, data, ...
- Dimensionality reduction required

Dimensionality Reduction

- Remove redundant, superfluous and harmful information
- Linear Dimensional Reduction
  - Principal Component Analysis (PCA)
  - Linear Discriminant Analysis (LDA)
- Nonlinear Dimensional Reduction
  - Kernel PCA, Multi-linear PCA, Kernel PCA
  - Maximum Variance Unfolding (semidefinite embedding), Isomap
  - **Multilayer Perceptrons (MLP), Bottleneck Features (BNF)**

Linear Discriminant Analysis



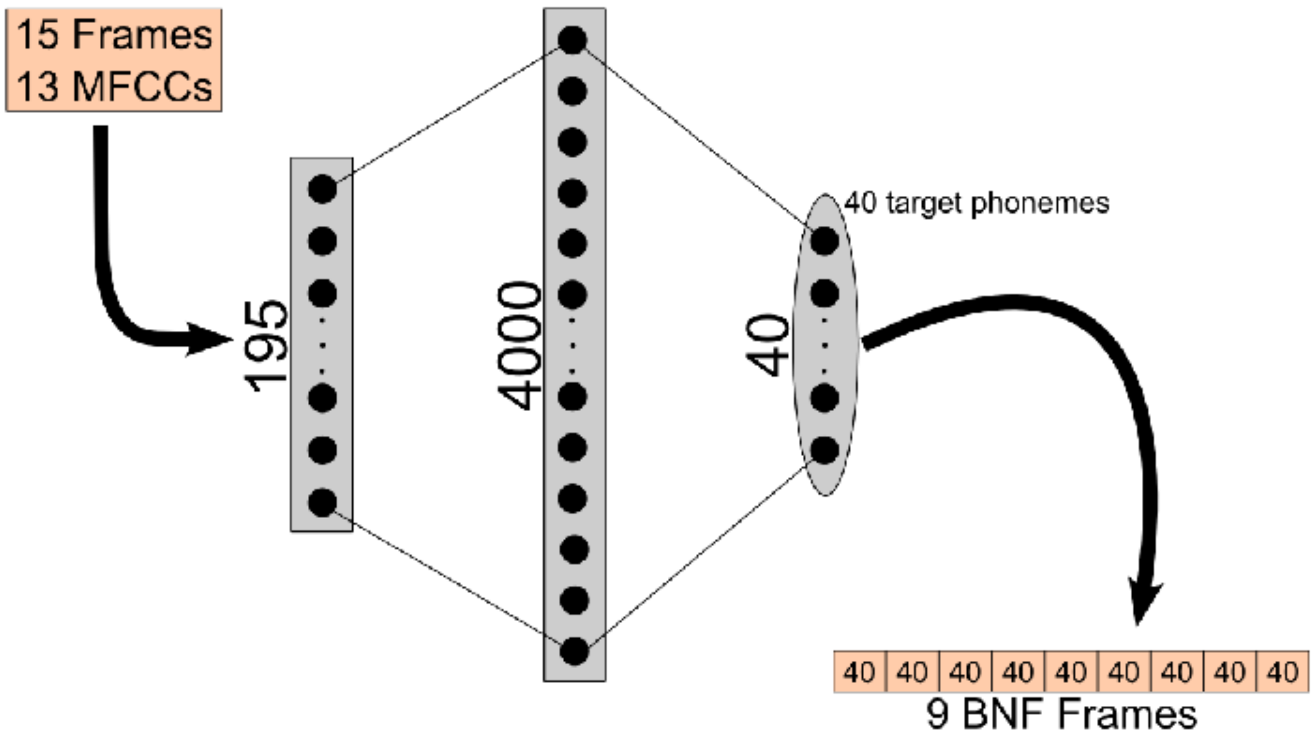
3dim LDA example: Image credits: Ivica Rogina

- Real World example:
- 20 dim features x 15 frame window
- => 300 input vectors
- 40 phonemes (target classes)
- desired output: 42 dim

MLP Features

- Input Layer: input vectors
- Output Layer: phonemes
- Hidden Layers: 1+ large hidden layers
- Learn MLP with back-propagation
- use output layer as feature vector
- Problem: reduced dim size same as #phonemes

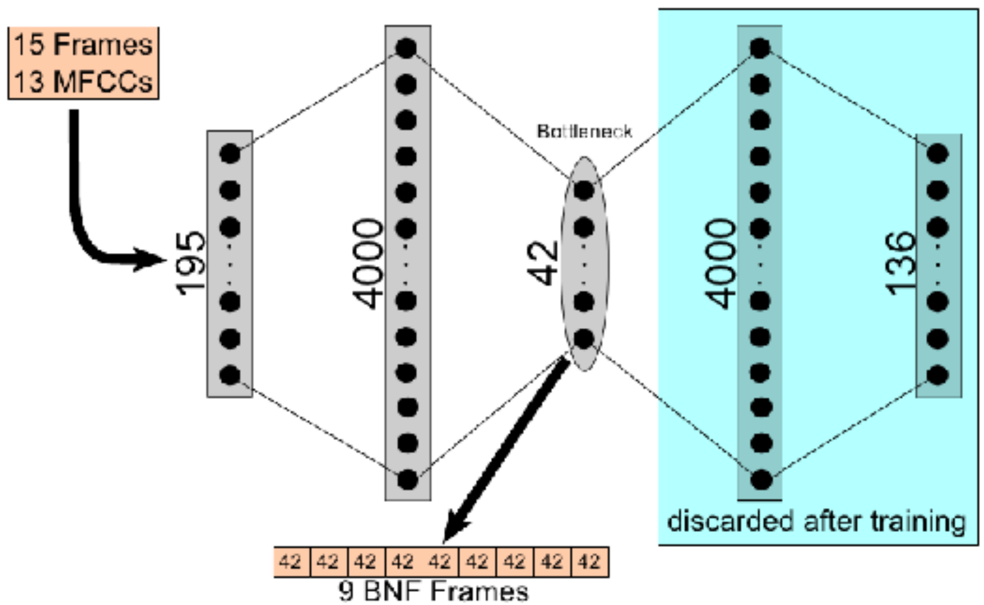
Bottleneck Features Example



Bottleneck Features

- Input Layer: input vectors
- Output Layer: phonemes (sub phonemes, phone-states)
- Hidden Layers: 2+ hidden layers
- Bottleneck Layer: small hidden layer
- Learn MLP with back-propagation
- use bottleck layer as feature vector

Bottleneck Features Example



The MLP architecture (4kx4k) that performed best in our experiments: A 15 frame context window, with 13 MFCCs each, was used as the input feature; the 136 node target layer (one node per sub-phone) and the 4k 3rd hidden layer were discarded after the MLP was trained. A 9 frame context window of the MLP output at the 42 node bottleneck layer is then used as the new 378 dim BNF feature. (LDA reduces the dimension to 42 again)

Bottleneck Features Evaluation

- MFCC Baseline: 20.04%
- MVDR Baseline: 19.95%
- final MFCC+MVDR system: 18.03%

Topo	EM Training	System Combination
2k	19.29%	17.27% / -
3k	18.99%	-
4k	18.99%	17.12% / 16.67%
2kx2k	19.10%	-
4kx4k	18.66%	- / 16.63%

Comparison of different bottleneck features. The EM Training column refers to a single BNF system trained to that stage. The System Combination column displays the WER of the final CNC of all 3 2nd pass systems, either self adapted or adapted on the CNC of the first pass.