Towards binaural modeling including cognition: the Two!Ears model

Hagen Wierstorf, Alexander Raake

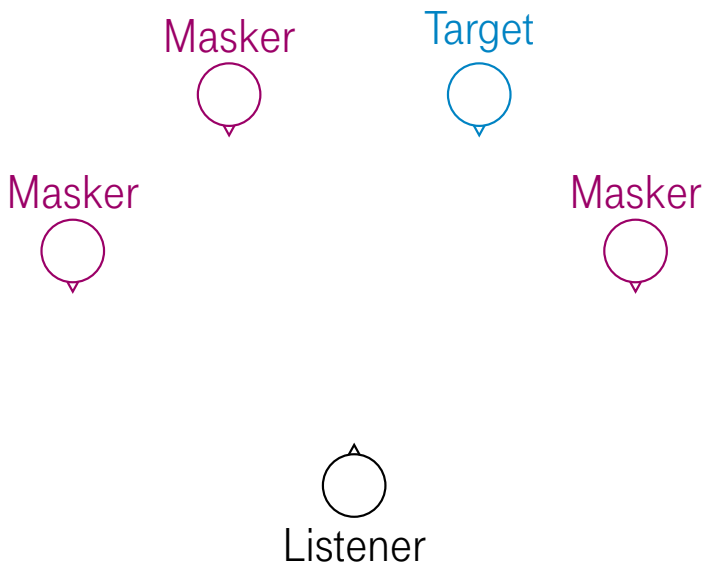
Institut für Medientechnik, TU Ilmenau

17. March 2016

TWO!EARS

TECHNISCHE UNIVERSITÄT ILMENAU

Motivation



Goal:

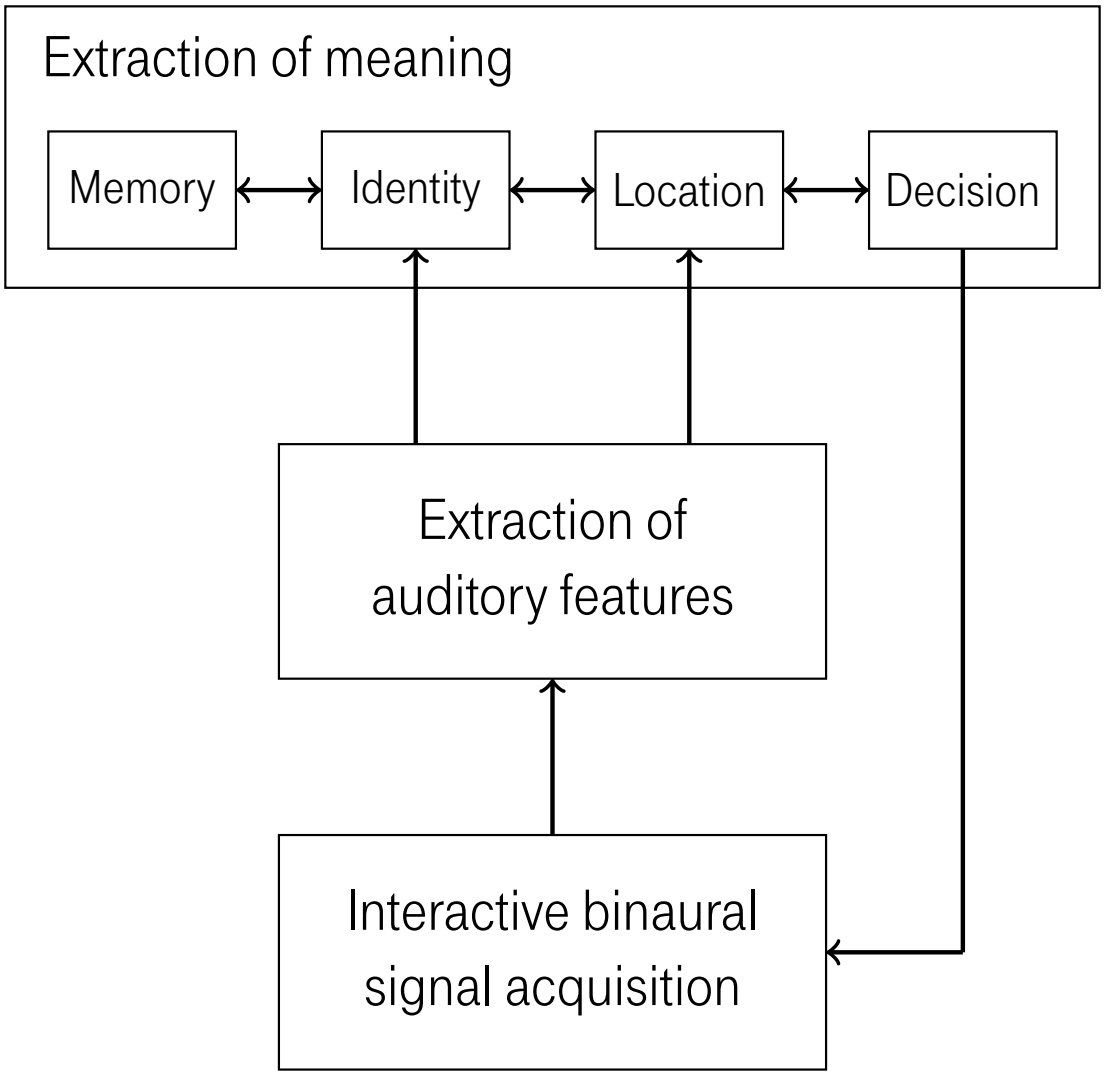
1. Identify target and localise it
2. Understand target

Results changes

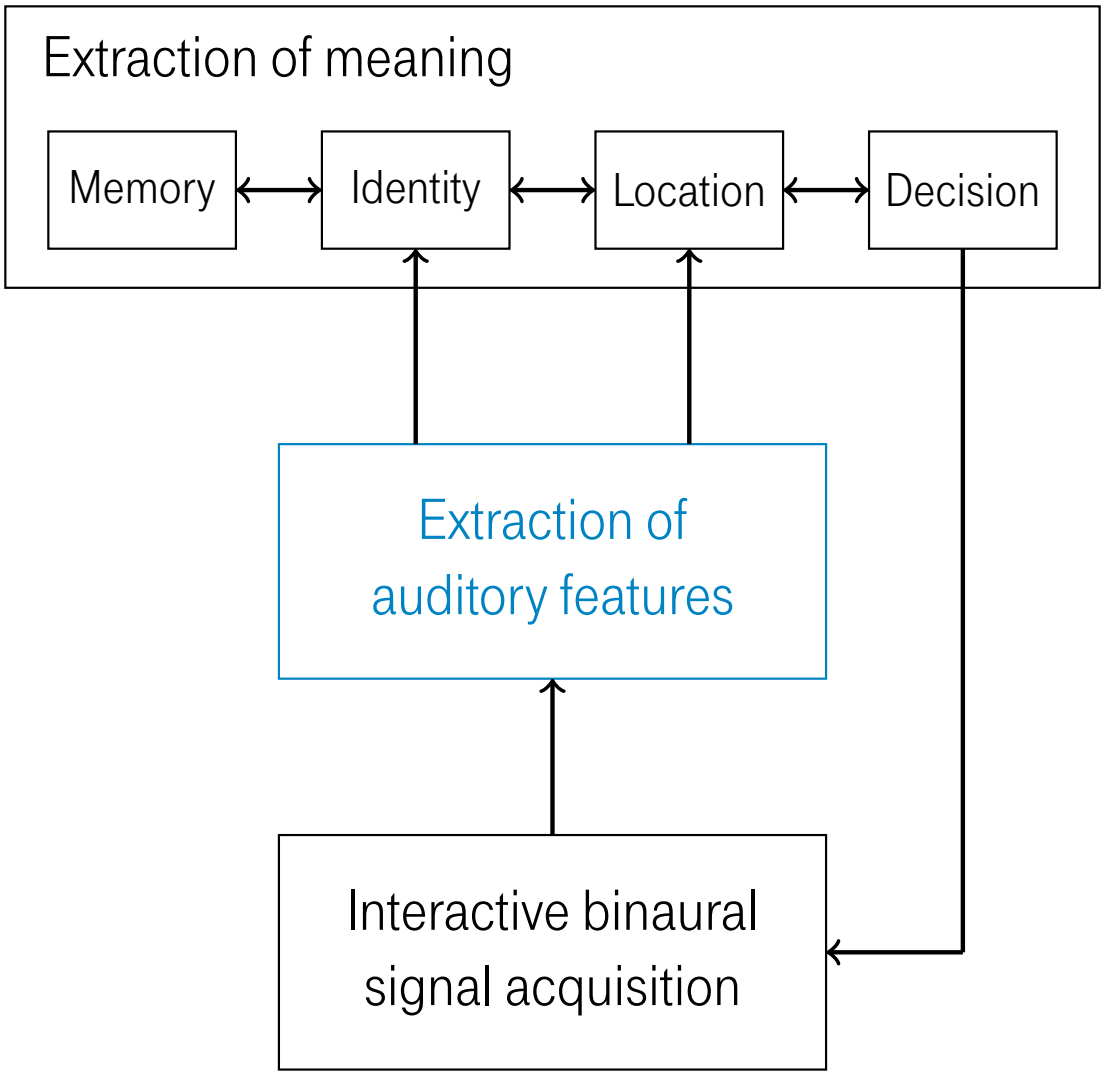
- Prior knowledge
- Interactive listening

Kopčo et al. (2010), Speech localization in a multitalker mixture, JASA
Brungart and Simpson (2007), Cocktail party listening in a dynamic multitalker environment, Perception & Psychophysics
Josupeit and Hohmann (2015), Modeling localization and word recognition in a multitalker setting, DAGA

Model structure



Auditory front-end



Auditory front-end

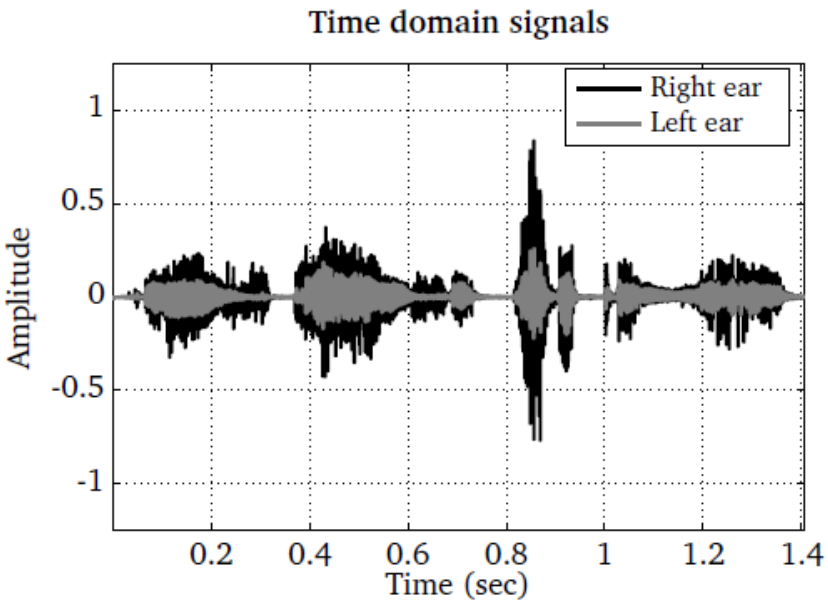
- AMToolbox, but in a combined manner
- Block based processing
- Change of parameter during processing
- Just ask for the auditory features you need

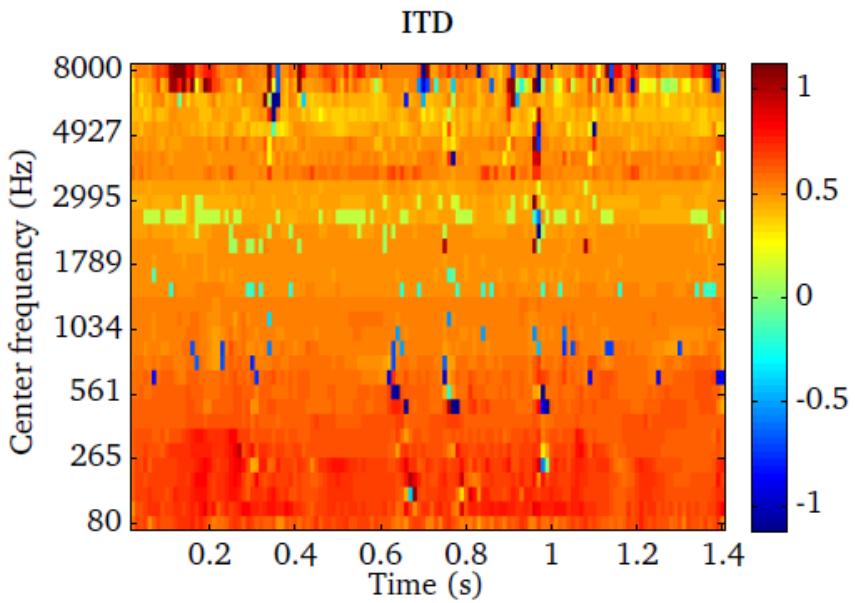
```
>> requestList

  Request name       Label                            Processor
  ------------       -----                            -----------------
  adaptation         Adaptation loop output           adaptationProc
  amsFeatures        Amplitude modulation spectrogram modulationProc
  autocorrelation    Autocorrelation computation      autocorrelationProc
  crosscorrelation   Crosscorrelation computation     crosscorrelationProc
  filterbank         DRNL output                      drnlProc
  filterbank         Gammatone filterbank output      gammatoneProc
  gabor              Gabor features extraction        gaborProc
  ic                 Inter-aural coherence            icProc
  ild                Inter-aural level difference     ildProc
  innerhaircell      Inner hair-cell envelope         ihcProc
  itd                Inter-aural time difference      itdProc
  moc                Medial Olivo-Cochlear feedback   mocProc
  myNewRequest       A description of my new request  templateProc
  offsetMap          Offset map                       offsetMapProc
  offsetStrength     Offset strength                  offsetProc
  onsetMap           Onset map                        onsetMapProc
  onsetStrength      Onset strength                   onsetProc
  pitch              Pitch estimation                 pitchProc
  precedence         Precedence effect                precedenceProc
  ratemap            Ratemap extraction               ratemapProc
  spectralFeatures   Spectral features                spectralFeaturesProc
  time               Time domain signal               preProc
```

Decorsière et al. (2015), Two!Ears Auditory Front-end 1.0, doi: 10.5281/zenodo.28008

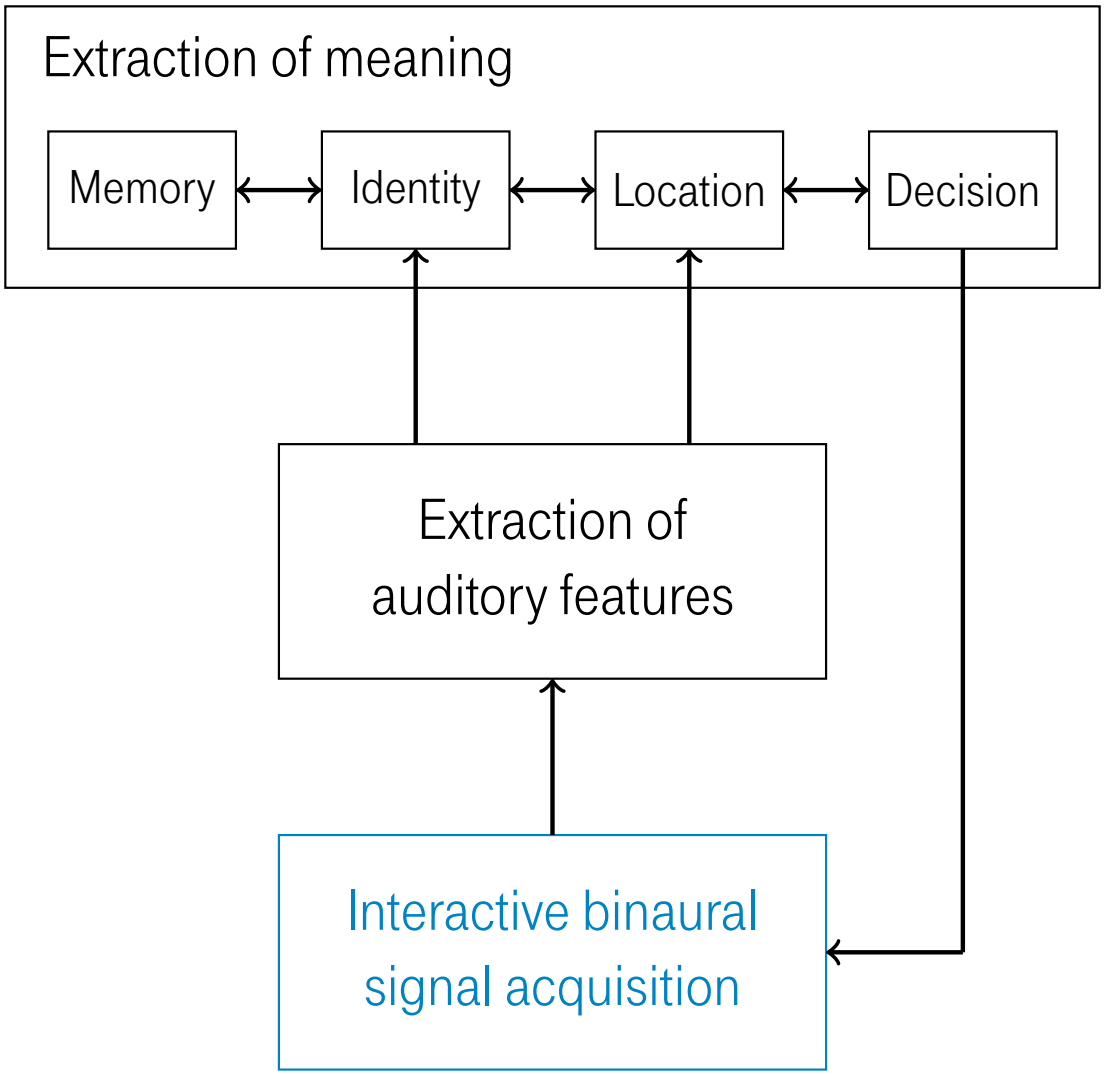
Auditory front-end

```
dObj = dataObject(earSignals(1:22494,:),fsHz);
requests = {'itd'};
parameter = genParStruct(...
    'fb_type',          'gammatone', ...
    'fb_lowFreqHz',     80, ...
    'fb_highFreqHz',    8000, ...
    'fb_nChannels',     32, ...
    'ihc_method',       'dau', ...
    'cc_wSizeSec',      0.02, ...
    'cc_hSizeSec',      0.01, ...
    'cc_wname',         'hann');
mObj = manager(dObj,requests,parameter);
mObj.processSignal();
dObj.plot([],[],'bGray',1,'decimateRatio',3,'bSignal',1);
dObj.itd{1}.plot;
```



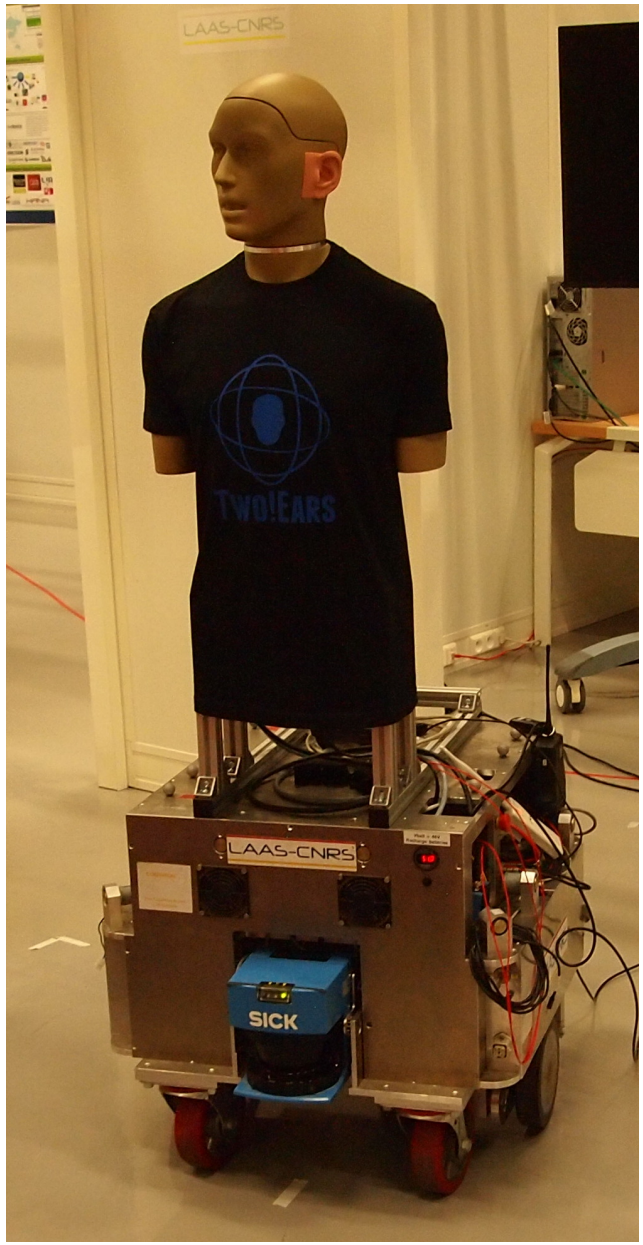


Robot / Binaural simulator



Robot

- Simple recording of binaural signals
- Allows for arbitrary positioning
- You need a robot
- Complicated software engineering



Bustamante et al. (submitted), Towards information-based feedback control for binaural active localization, ICASSP

Binaural simulator

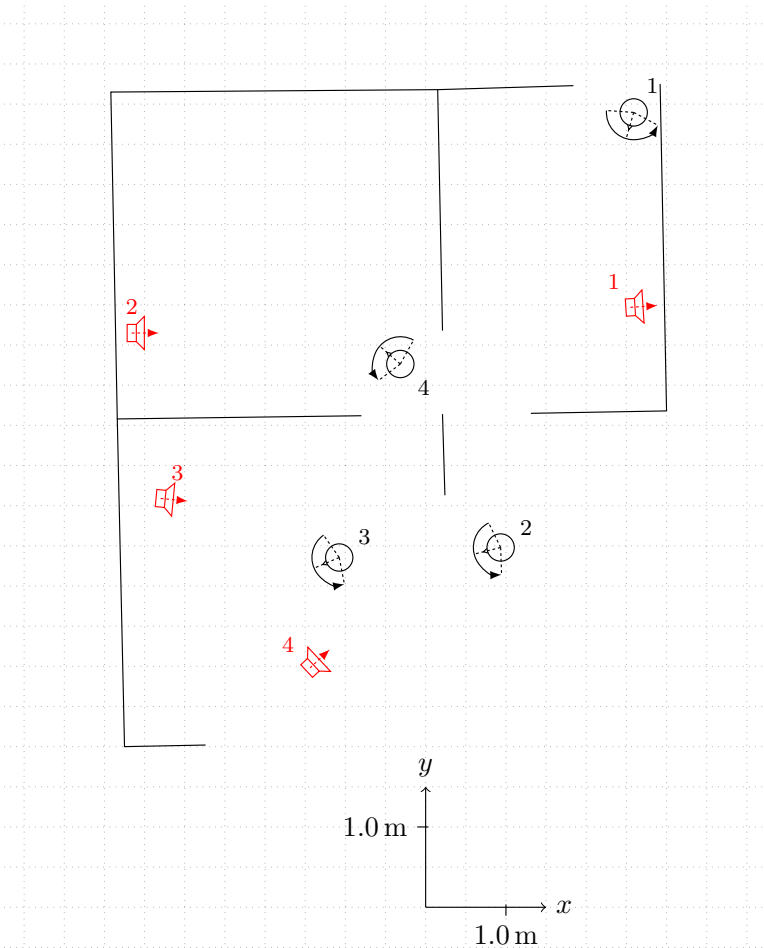
- Block-based convolution of impulse responses and audio material
- Uses the convolution C++ core of the SoundScape Renderer ⇒ mex-file
- Acoustic scene has to be specified
- Database needed

```
<scene
  BlockSize="4096"
  SampleRate="44100"
  MaximumDelay="0.05"
  PreDelay="0.0"
  LengthOfSimulation="5.0"
  NumberOfThreads="1"
  Renderer="ssr_binaural"
  HRIRs="impulse_responses/qu_kemar_anechoic/QU_KEMAR_anechoic_3m.sofa">
  <source Position="1 2 1.75"
          Type="point"
          Name="Cello"
          Volume="0.4">
    <buffer ChannelMapping="1"
            Type="fifo"
            File="stimuli/anechoic/instruments/anechoic_cello.wav"/>
  </source>
  <sink Position="0 0 1.75"
        UnitX="1 0 0"
        UnitZ="0 0 1"
        Name="Head"/>
</scene>
```

Winter et al. (2015), Two!Ears Binaural Simulator 1.0, doi:10.5281/zenodo.28010

Binaural simulator

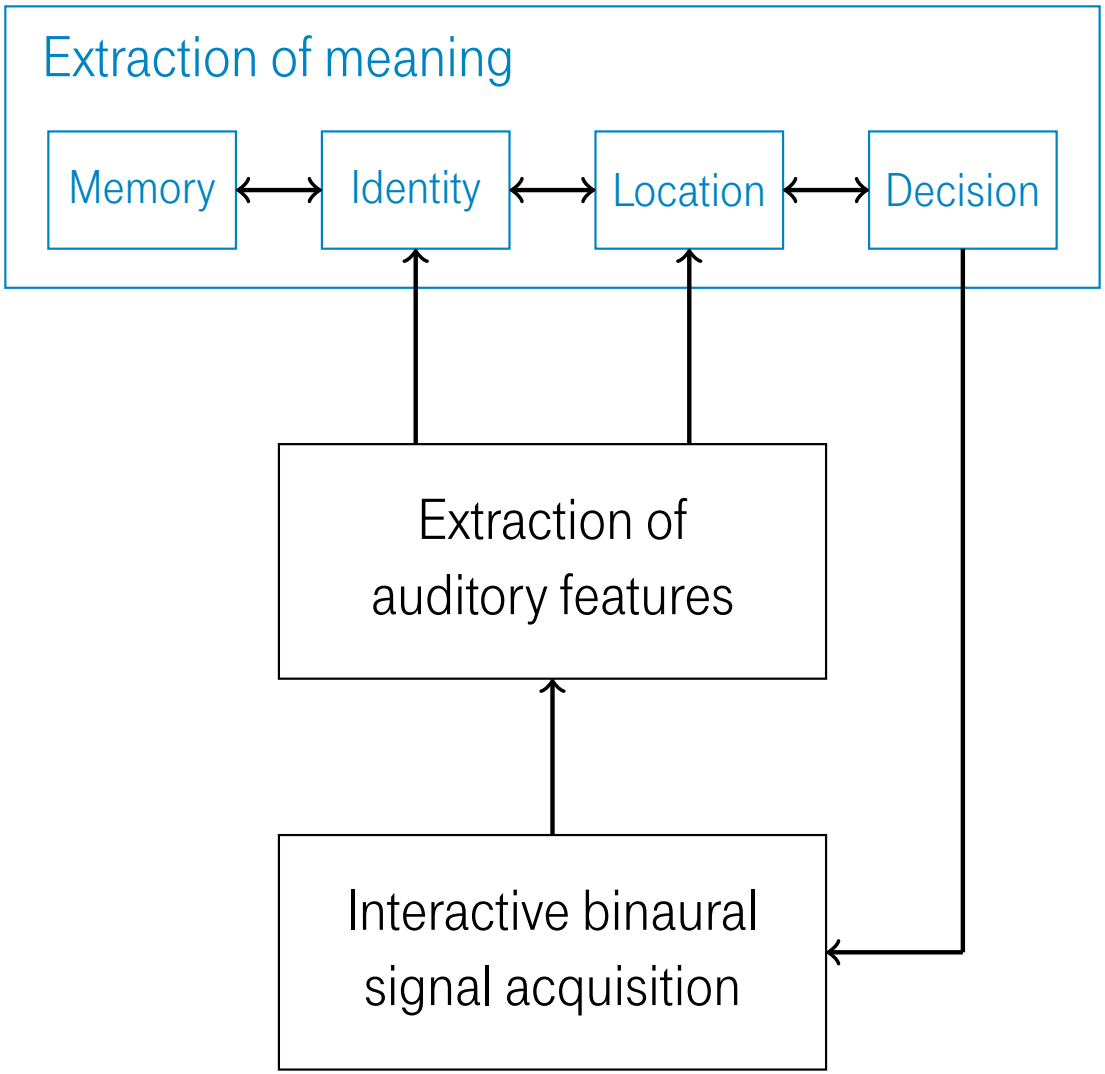
- Database of impulse responses
- Collection of new measurements and existing ones
- Usage of SOFA file format



Loudspeaker and KEMAR positions

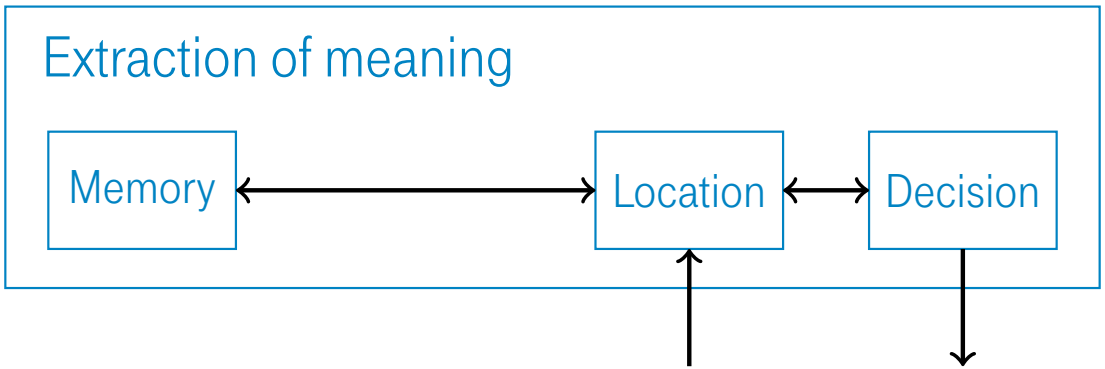
Winter et al. (submitted), Database of binaural room impulse responses of an apartment-like environment, 140th AES

Blackboard system



Blackboard system

Localization of multiple sources in reverberant environments



Performance increases by

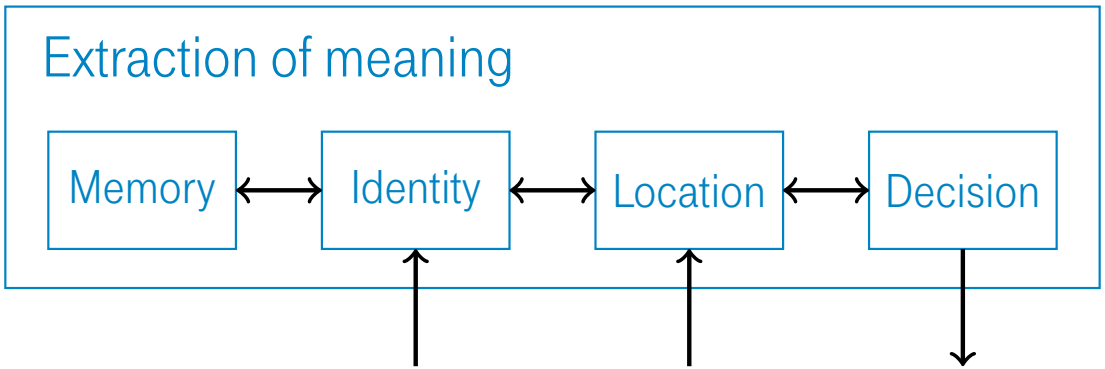
- Multi-conditional training
- Step wise head rotations

Ma et al. (2015), A machine-hearing system exploiting head movements for binaural sound localisation in reverberant conditions, ICASSP

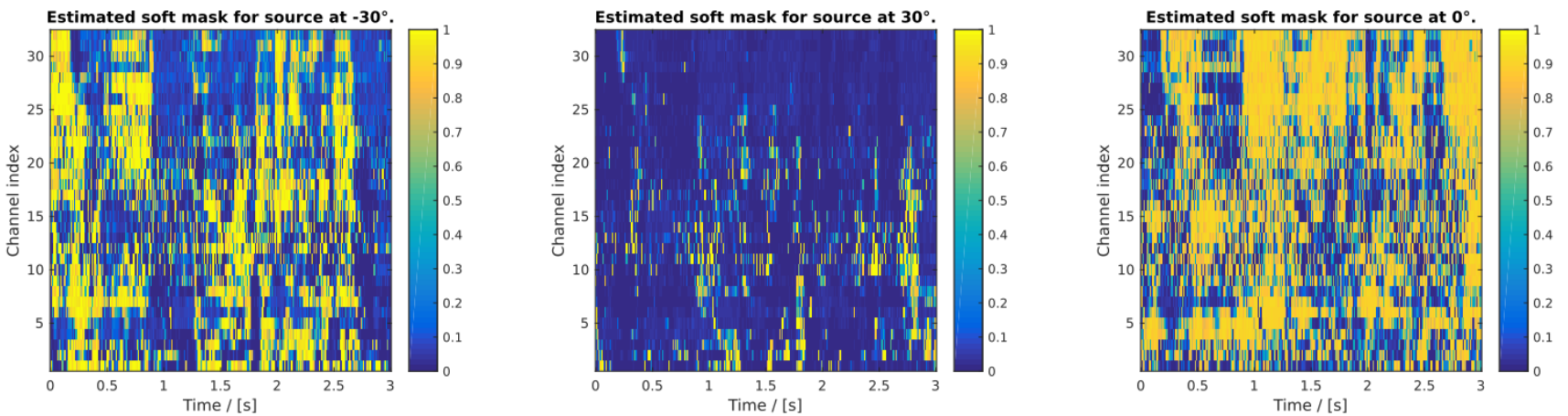
May et al. (2015), Robust localisation of multiple speakers exploiting head movements and multi-conditional training of binaural cues, ICASSP

Blackboard system

Identify target and localize it



Interaction between localisation and identification implemented by segmentation:

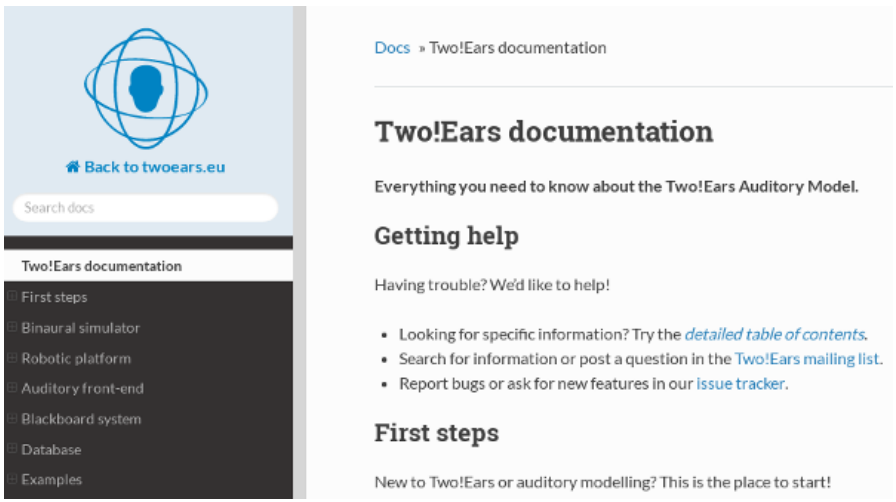


Ma et al. (2015), Exploiting top-down source models to improve binaural localisation of multiple sources in reverberant environments, Interspeech

Getting involved

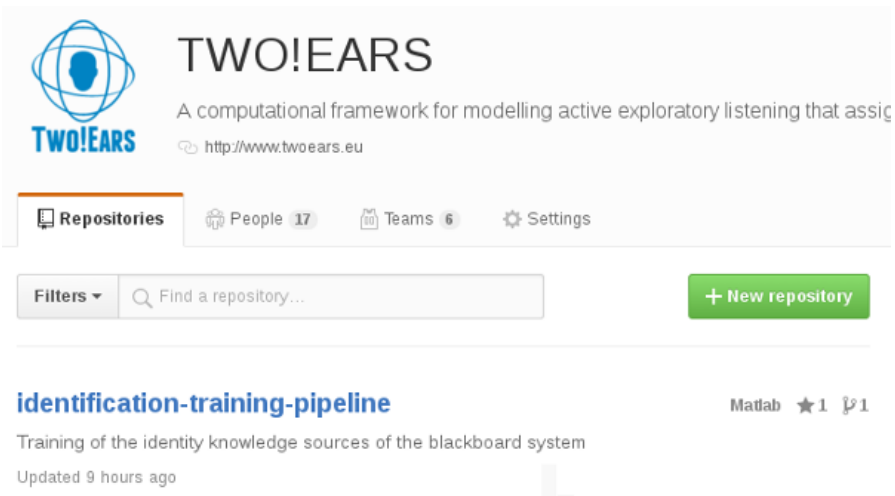
Ultimate Goal is to provide a framework that can be used by everyone in order to help advance binaural modeling

Documentation



`http://twoears.aipa.tu-berlin.de/doc`

Development



`https://github.com/twoears`

`http://twoears.eu`

Conclusion

Highlights:

- Incorporation of top-down processes
- Auditory front-end: just ask for an auditory feature
- Binaural simulator: interaction with the acoustic scene
- Database: large collection of HRIRs and BRIRs all in the same format
- Large documentation

Challenges:

- Complexity of the model
- Usability could be improved


http://spatialaudio.net