An Overview of MPEG-4 Audio Version 2

Heiko Purnhagen

Laboratorium für Informationstechnologie
University of Hannover, Germany
Outline

- Introduction
- Review of MPEG-4 Audio Version 1
- New Tools of MPEG-4 Audio Version 2
- Outlook
- Conclusions
Introduction: MPEG-4

● New multimedia applications demand ...
  – efficient & flexible coding of natural & synthetic audiovisual content

=> Development of MPEG-4 Standard: "Coding of audiovisual objects"

● MPEG-4 Audio Version 1  (Oct. 1998)
  – audio and speech coding @ 2 .. 64+ kbit/s/ch

● MPEG-4 Audio Version 2  (Dec. 1999)
  – additional functionalities
Introduction: MPEG-4 Versions

- **Problem:** Tight schedule for MPEG-4
  => several promising tools not mature in time

- **Solution:** MPEG-4 Version 2 (Amendment 1)
  - backward compatible extension of Version 1
  - new tools added => additional functionalities
**MPEG-4 Audio Requirements:**
- efficient coding (various content types / bit rates)
- other functionalities (e.g. scalability)

=> Combination of coding techniques required
- utilise different source and perception models
Version 1: Tools

- **MPEG-4 Version 1**
  - Audio Tools: **coding** of audio objects

<table>
<thead>
<tr>
<th></th>
<th>Speech</th>
<th>General Audio</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Natural</strong></td>
<td>• HVX C (param.)</td>
<td>• TwinVQ (6 .. 16 kbit/s/ch)</td>
</tr>
<tr>
<td></td>
<td>(2 .. 4 kbit/s)</td>
<td>• AAC (+scalable)</td>
</tr>
<tr>
<td></td>
<td>• CELP (NB+WB)</td>
<td>(16 .. 64+ kbit/s/ch)</td>
</tr>
<tr>
<td></td>
<td>(4 .. 24 kbit/s)</td>
<td></td>
</tr>
<tr>
<td><strong>Synthetic</strong></td>
<td>• TTS-Interface</td>
<td>• SA (incl. MIDI)</td>
</tr>
</tbody>
</table>

- Systems Tools: **composition** of audio objects
  => audio scene (mixing, effects: **Structured Audio**)
Version 2: Overview

- **New Audio Tools**
  - Error Robustness
  - Low-Delay Audio Coding
  - Small Step Scalability
  - Parametric Audio Coding
  - CELP/HVXC Silence Compression

- **New Systems Tools**
  - Environmental Spatialisation
  - MP4 File Format
Version 2: Error Robustness

- Goal: Transmission on error-prone channels
- Approach: Error Robustness Tools
  - Error Protection tool
  - Error Resilience for source coding tools
Version 2: Error Robustness

- **Error Protection Tool**
  - Unequal Error Protection: error sensitivity classes
  - **Cyclic Redundancy Check / Forward Error Correction**

![Diagram of Error Protection Tool](image-url)
Version 2: Error Robustness

- Error Resilience for source coding tools
  - Error Resilience for AAC
e.g. Huffman Codeword Reordering
  => less error propagation
  - Error Resilience for other coding tools
  => reduced detoriation in case of bit errors

- Error Concealment in decoder (not normative)
Version 2: Error Robustness

- **Demo:** AAC 48 kHz mono @ 64 kbit/s random bit error rate: $10^{-3}$
  - no Error Robustness (frame sync only)
  - Error Protection, Resilience, and Concealment (19% mean bit rate overhead)
Version 2: Low-Delay Audio Coding

● Goal: Low-delay general audio coding
  – realtime bi-directional communication
  – AAC algorithmic delay:
    e.g. 24 kHz @ 24 kbit/s
    => 110 ms + max. 210 ms (bit reservoir)

● Approach: Low-Delay Audio Coder
  – derived from AAC
**Version 2: Low-Delay Audio Coding**

- **Low-Delay Audio Coder (modified AAC)**
  - frame length & filterbank delay
    => 1/2 window size
  - "look-ahead" for window-switching
    => **Zero-Padded** window for transients
  - bit reservoir
    => not used

=> 20 ms algorithmic delay (48 kHz)
Version 2: Small Step Scalability

● Goal: Small step bit rate scalability
  – AAC scalability: typ. 16 kbit/s enhancement layers

● Approach: Bit-Sliced Arithmetic Coding
  – combined with AAC:
    BSAC replaces AAC Huffman coding
    => 1 kbit/s/ch enhancement layers
  – Principle:
    transmit bit-slices with most significant bits first
    enhancement: less significant bits (finer quant.)
    higher frequency bands
Version 2: Small Step Scalability

**Demo:** BSAC 48 kHz mono @ 64 kbit/s
Goal: Very low bit rate audio coding
- source model for music?

Approach: Parametric Signal Representation
- Signal decomposition into components: "Harmonic and Individual Lines plus Noise" (HILN)
- Functionalities:
  - very low bit rate (4 .. 16 kbit/s)
  - speed and pitch change (decoder)
  - bit rate scalability
Version 2: Parametric Audio Coding

Parametric Audio Encoder (HILN)
Version 2: Parametric Audio Coding

**Demo:** HILN 16 kHz mono @ 6 kbit/s

Original Signal

Parameter Decoding
- Dequant
- Dequant
- Dequant

Synthesis
- Harmonic Components
- Sinusoidal Components
- Noise Components

Audio Signal

Parametric Audio Decoder (HILN)
Version 2: Silence Compression

- **Goal:** Bit rate reduction if no speech activity
  - bi-directional communication: ~50% silence

- **Approach:** HVXC/CELP Silence Compression
  - generate comfort noise if no speech activity

![Diagram of Silence Compression System](image-url)
Version 2: Environmental Spatialisation

- **Goal:** Efficient and flexible "3-D" audio scenes
  - SA-based composition not easy for "3-D" scenes

- **Approach:** Physical or perceptual description

![Graph showing typical room response with direct sound, early reflections, room effect, late reverberation, and time axis.](image_url)
Version 2: Environmental Spatialisation

● Physical approach
  – description of acoustical properties of environment (room geometry, sound source position, ...)
  – corresponding audio and visual scene e.g. 3-D virtual reality

● Perceptual approach
  – high-level perceptual description of "audio scene" (room reverberance, source presence, ...)
  – audio and visual scene independent e.g. movie-like applications
Version 2: Other Tools

- MPEG-4 File Format (MP4)
  - flexible format for:
    - interchange, editing, presentation
  - based on QuickTime

- Backchannel
  - e.g. adaptive streaming
Outlook

- Version 2 Profiles and Levels
  - currently under discussion

- Version 2 Verification Test
  - scheduled for autumn 1999

- Encoder Optimisation (not normative)
  - e.g. automatic segmentation of speech / music
    (audio objects are transparent ...)

Hannover
Conclusions

● MPEG-4 Audio Version 2
  – backward compatible extension of Version 1
  – additional functionalities by new tools

● MPEG-4 Version 2 finalised: Dec. 1999

● New work item: MPEG-7
  "Multimedia Content Description Interface"
Conclusions

- MPEG Audio Web Page
  http://www.tnt.uni-hannover.de/project/mpeg/audio/

- Further Demonstrations
  - Error Robustness (for AAC)
  - Low-Delay Audio Coding (realtime en-/decoder)
  - Small Step Scalability (BSAC)
  - Parametric Audio Coding (HILN)