A black and white logo

Description automatically generated ISO/IEC JTC 1/SC 29/WG 2 N453

**ISO/IEC JTC 1/SC 29/WG 2**

**MPEG Technical requirements   
Convenorship: SFS (Finland)**

**Document type:** Output Document

**Title:** Use cases and requirements on audio coding for machines

**Status:** Approved

**Date of document:** 2025-04-04

**Source:** ISO/IEC JTC 1/SC 29/WG 2

# Expected action: none

**Action due date:** none

**No. of pages:** 16 (without cover page)

**Email of Convenor:** igor.curcio@nokia.com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg2>

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 2**

**MPEG TECHNICAL REQUIREMENTS**

**ISO/IEC JTC 1/SC 29/WG 2 N453**

**Online – April 2025**

**Title: Use cases and requirements on audio coding for machines**

**Source: WG 2 MPEG Technical Requirements**

**Status: Approved**

**Serial Number: 25076**

# Abstract

This document explains the scope, the use cases and requirements of the exploration item “Audio Coding for Machines (ACoM)”.

# Introduction

Traditional coding methods aim for the best audio under certain bit-rate constraint for human consumption. However, with the rise of machine learning applications, along with the abundance of sensors, many intelligent platforms have been implemented with massive data requirements including scenarios such as connected vehicles, audio surveillance, machine diagnostics and smart city. In many of these applications the spatial distribution of audio contains important information. Medical data, like those coming from EEG and EKG measurements, while not inherently audio, often have a similar structure and similar requirements as audio data. Spatial audio and medical data will be referenced as multi-dimensional streams.

The sheer quantity of data being produced constantly leads previous methods with a human in the pipeline to be inefficient, and unrealistic in terms of latency and scale. There are additional concerns in transmission and archive systems which require a more compact data representation and low latency solution. This led to the introduction of Audio Coding for Machines.

In some cases, machines will communicate amongst themselves to perform tasks without a human in the mix, while in others there will be a need for additional human consumption of the specific decompressed stream. This specific scenario is possible in surveillance use cases, where a human “supervisor” may occasionally search for a specific sound, or scene in the audio. In other cases, the corresponding bitstream may be used for both human and machine consumption. In the case of cars, the features may be used to overcome the good sound insulation of modern cars (“detection of ambulance sirens”) and for monitoring failure of components (including predicted maintenance).

Any use cases in which audio features need to be transmitted for additional processing which may potentially be used for machine or human end users could benefit from a standard in the coded features (shared backbone). Interoperability is crucial where different manufacturers and platforms need communication to achieve a common goal.

Additionally, the feature stream must be efficient for both transmission and archive concerns for both latency and space. A standard for the compressed coding of this feature stream will establish an efficient protocol for machines to communicate.

* 1. **Scope**

MPEG-ACoM aims to define a bitstream and data format for compressing audio, multi-dimensional streams, or features extracted from such signals that is efficient in terms of bitrate/size and can be used by a network of machines after decompression to perform multiple tasks without significantly degrading task performance. The decoded audio, multi-dimensional streams or features can be used for machine consumption or hybrid machine and human consumption. In addition to the essence the format must also contain metadata describing how the data audio or multi-dimensional stream was captured.

The first phase will be application agnostic: Data is encoded near-lossless enabling the training of feature extraction schemes. The result from this phase is already useful for industry simplifying the exchange of data using this standardized format.

In a second phase feature extraction schemes are added. These features will be optimized for different applications.

* 1. **System Overview**

The generic system architecture contains a pair of ACoM encoder and decoder. The input of the ACoM system could be metadata describing the input and either of

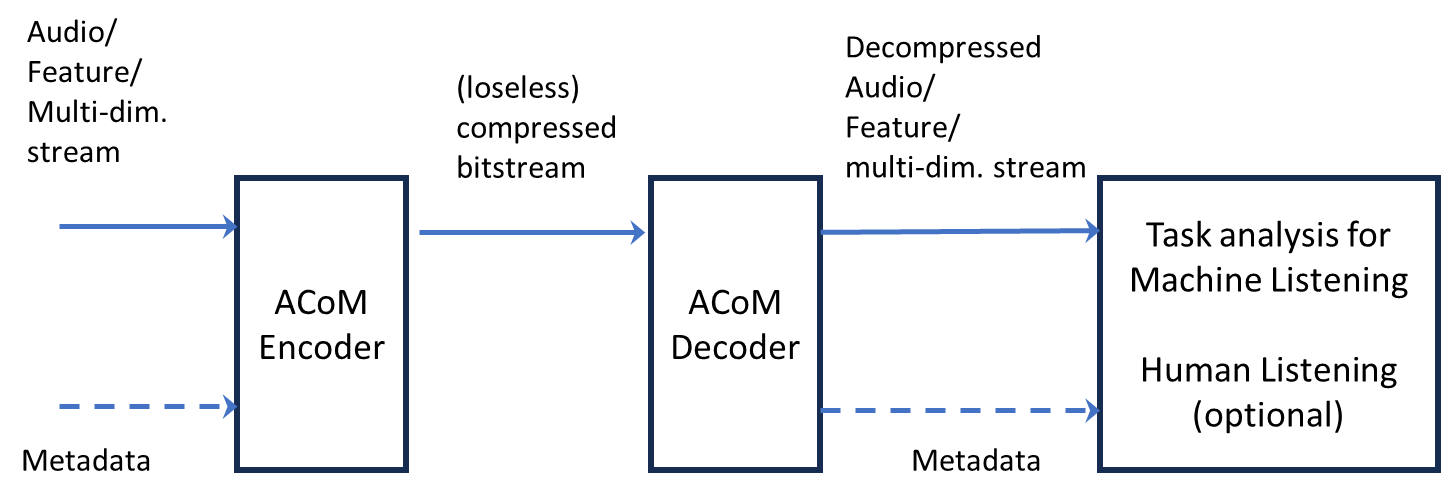
* audio signals (one or multi-dimensional)
* multi-dimensional streams (e.g. medical data)
* extracted features (phase 2 only).

In case of a feature stream, the type and format of feature will be specified, features may take different forms depending on the application. Feature extraction and coding is not in phase 1 but in phase 2.

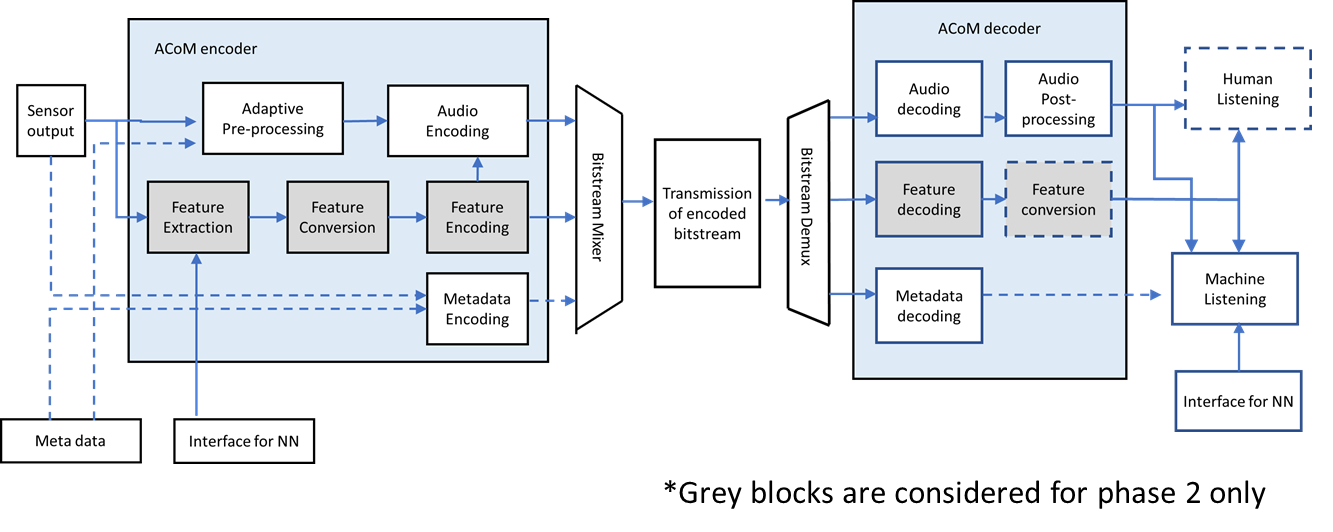
The decompressed bitstream of audio and/or multi-dimensional streams and/or features may then be used for post-processing tasks, which may include machine consumption tasks or hybrid machine and human consumption tasks. The encoder can be optimized for either a single task or multiple, and the size of the compressed stream and compare favorably to traditional coding techniques on the unprocessed audio or medical data.

The MPEG activity on Audio Coding for Machines (ACoM) aims to standardize a bitstream format generated by compressing a previously extracted feature stream or data stream. Compression scheme of data stream including the audio and metadata at each phase are followed:

|  |  |  |
| --- | --- | --- |
| Phase | Data Type | Compression Scheme |
| Phase 1 | Audio/Signal | Lossless compression |
| Metadata | Lossy/lossless compression |
| Phase 2 | Audio/Signal | Lossy/lossless compression |
| Metadata | Lossy/lossless compression |



**Figure 1:** Pipeline for ACoM: Phase 1 using lossless compression. In Phase 2, compression is expected to use (lossy) feature compression.



**Figure 2**: An example of potential ACoM architecture (both phase 1 and phase 2).

Fig 2 shows an example of potential ACoM architecture. The ACoM codec could be an audio codec or a feature codec, or both. In case of a feature codec, the ACoM feature encoding is consisting of feature extraction, feature conversion and feature coding. In case of an audio codec, an adaptive pre-processing may be included in ACoM encoder, and an adaptive post-processing may be included in ACoM decoder. Metadata may be fed into both audio and metadata encoding modules. There may be an interface to an external neural network (NN) for the feature extraction and the task specific networks.

# Use Cases

|  |
| --- |
| * 1. **UC1 Predictive Maintenance** |
| Target phase: Phase 1 |
| Acoustic sensors are used in industrial environments to monitor the function of a single machine or all machines in a hall. Predictive maintenance is replacing components before failure but as late as possible. Acoustic sensors can be outside a machine. Therefore, retrofit of old machines is possible. Neural networks are used for training detectors. Usually, the difference to be detected are small compared to differences within the data in normal operation. Usually, there is much more training data available for correct operation then for failure conditions. |
| **Overlap with other use cases** |
| UC2 Process control, UC3 in-line testing and UC4 end-of-line testing |
| **Required properties of the algorithm** |
| * Capable to encode many microphone channels around a machine * Capable to encode metadata (see sub-tasks) * The compression must be near-lossless to adapt to many different types of machines and failures * The format shall contain time stamps. This ensures a phase relationship between the individual single or multichannel source streams. Note: Synchronized clocks of the sources are ensured by other means. See also UC6 |
| **Optional properties of the algorithm (Need further refinements)** |
| * In some cases, it might be feasible to auralize the reconstructed audio for human inspection. * A block mode facilitates data storage |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene exploiting redundancies of measurement signals * Coding of metadata describing type and position of sensors * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Coding of process steps (material, tools) stored in the data file * Coding of operation condition (at least OK/NOK, in case of NOK optionally different type of predicted failure) * Source separation through remote sensor lines * Capturing of non-audible sound (as well as in UC8) * Coding through sensor nodes with limited computing power |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth might be larger since the systems are stationary. The limitation is the amount of storage. * Near real-time processing is necessary. * Power is not a large concern, however see sub-task |

|  |
| --- |
| * 1. **UC2 Process Control** |
| Target phase: Phase 1 |
| Acoustic sensors are used in industrial environments to control the function of a machine. Process parameters are modified to compensate aging of components or to adapt to differences in material processed. Acoustic sensors can be outside a machine and therefore not influenced by dirt or other disturbance. Neural networks are used for to train process parameters. |
| **Overlap with other use cases** |
| UC1 Predictive Maintenance, UC3 In-Line Testing and UC4 End-of-Line Testing |
| **Required properties of the algorithm** |
| * Capable to encode multiple microphone channels close to the machine. * Capable to encode metadata (see sub-tasks). * The compression must be near-lossless to adapt to many different types of machines. |
| **Optional properties of the algorithm** |
| * In some cases, it might be useful to code operation condition (OK/NOK). * Auralisation not necessary. |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene exploiting redundancies of measurement signals. * Coding of metadata describing type and position of sensors, machines and of the room acoustics. * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Coding of process steps (material, tools) stored in the data file |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth might be larger since the systems are stationary. * Real-time processing is necessary. * Power is not a large concern. |

|  |
| --- |
| * 1. **UC3 In-line Testing** |
| Target phase: Phase 1 |
| Acoustic sensors are used in industrial environments to detect whether individually produced components are out of spec. Such components can be discarded before further processing. Acoustic sensors are close to the process in the machine. Neural networks are used for to train detectors. Usually, the difference to be detected are small. Usually, there is a lot of environmental noise. Usually, there is much more training data available for correct operation then for failure conditions. |
| **Overlap with other use cases** |
| UC1 Predictive Maintenance, UC2 Process Control, and UC4 End-of-Line Testing |
| **Required properties of the algorithm** |
| * Capable to encode multiple microphone channels specially to discard noise * Capable to encode metadata (see sub-tasks) * The compression must be near-lossless to adapt to many different types of machines and failures |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene exploiting redundancies of measurement signals * Coding of metadata describing type and position of sensors * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Coding of process steps (material, tools) stored in the data file * Coding of operation condition (at least OK/NOK, in case of NOK optionally different types of predicted failures) |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth might be larger since the systems are stationary. The limitation is the amount of storage. * Real-time processing is necessary. * Power is not a large concern. |

|  |
| --- |
| * 1. **UC4 End-of-line Testing** |
| Target phase: Phase 1 |
| Acoustic sensors are used in industrial environments to detect whether a final product is out of spec. Acoustic sensors are after the process in the machines and might be in a sound insulated measurement cabin. Neural networks are used to train detectors. Usually, the differences to be detected are small. Usually, there is no environmental noise enabling very precise measurement. Usually, there is much more training data available for correct operation then for failure conditions. |
| **Overlap with other use cases** |
| UC1 Predictive Maintenance, UC2 Process Control and UC3 In-line Testing. |
| **Required properties of the algorithm** |
| * Capable to encode multiple microphone channels * Capable to encode metadata (see sub-tasks) * The compression must be near-lossless to adapt to many different types of products and failures |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene exploiting redundancies of measurement signals * Coding of metadata describing type and position of sensors * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Coding of name of product stored in the data file * Coding of condition (at least OK/NOK, in case of NOK optionally different types of predicted failures) |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth might be larger since the systems are stationary. The limitation is the amount of storage. * Real-time processing is necessary. * Power is not a large concern. |

|  |
| --- |
| * 1. **UC5 Traffic Monitoring and Control** |
| Target phase: Phase 2 |
| Acoustic sensors are used to monitor traffic flow. For this purpose, a network of acoustic sensors is installed. The sensors could not only count cars but also classify them into groups and control traffic signs by detecting siren enabling faster advancement of emergency vehicles. The acoustic sensor network can also be used to track crowds of people in cities. |
| **Overlap with other use cases** |
| UC6 Construction Site Monitoring. |
| **Required properties of the algorithm** |
| * Capable to encode a few microphone channels in each sensor node * Capable to encode metadata (see sub-tasks) |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene * Edge computing in each node * Coding of metadata describing type and position of sensors and exact time * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Optional privacy compliance: Automatic detection and suppression of (unwanted) speech recordings and/or lossy coding to avoid possibility of auralisation. |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth shall be low because sensor node connection might be wireless. * Near-Real-time processing is necessary. * Power shall be low if the nodes are battery powered. |

|  |
| --- |
| * 1. **UC6 Construction Site Monitoring** |
| Target phase: Phase 2 |
| Acoustic monitoring is used for automatic detection and tracking of vehicles entering in and driving thru large construction sites. This enables monitoring the flow of goods, preventing accidents by predicting conflicts, and to avoid stealing goods. In addition, such systems can be used to identify the guilty party in case of noise pollution above legal restrictions.  Construction sites are usually dirty. Acoustic sensors are robust and can even work when partially occluded by objects. |
| **Overlap with other use cases** |
| UC5 Traffic Monitoring and Control. |
| **Required properties of the algorithm** |
| * Capable to encode a few microphone channels in each sensor node * Capable to encode metadata (see sub-tasks) * All captured data must be time-stamped to track individual vehicles. * The algorithm must be capable to detect position of vehicles and intruders to enable alarms for dangerous positions. |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene * Edge computing in each node * Coding of metadata describing type and position of sensors and exact time * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Optional privacy compliance: Automatic detection and suppression of (unwanted) speech recordings and/or lossy coding to avoid possibility of auralisation. |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth shall be low because sensor node connection might be wireless. * Near-Real-time processing is necessary. * Power be low if the nodes are battery powered. |

|  |
| --- |
| * 1. **UC7 Speech Recognition and Acoustic Scene Analysis** |
| Target phase: Phase 1 |
| Speech Recognition and Acoustical Scene Analysis is used as user interfaces and/or for improving speech in noise environments. Neural networks for such applications need training data:   * Training data for speech recognition * Training data for speaker recognition * Training data for adaptive, real-time enhancement of speech in noisy situations * Training data for sentiment analysis * Training data for emotion recognition * Training data for environment sound analysis |
| **Overlap with other use cases** |
| UC12 Artistic creation |
| **Required properties of the algorithm** |
| * Capable to encode multiple microphone channels * Capable to encode metadata (see sub-tasks) |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing the whole acoustic scene * Coding of metadata describing type and position of sensors and exact time * Coding of metadata describing measurement conditions (e.g. temperature, humidity, air pressure) * Coding of metadata describing audio transcription * Metadata for annotation of content, e.g. language, speaker, spoken word |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Bandwidth is not an issue because there is no transmission of content * Real-time processing is necessary. * Power shall be low because devices are battery powered. |

|  |
| --- |
| * 1. **UC8 Timed Medical Data** |
| Target phase: Phase 2 |
| Timed Medical Data like EEG and ECG consist of several one-dimensional streams and these are very similar to the data captured by microphones. Long-time monitoring and multi-electrode measurement create huge data sets. Currently, exploitation of measurements of many patients is limited due to storage constraints and privacy issues. |
| **Overlap with other use cases** |
| None |
| **Required properties of the algorithm** |
| * Capable to encode a few channels * Capable to encode metadata (see sub-tasks) * Flexible sampling rates |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing and near-lossless coding of all streams exploiting redundancies * Capturing of non-audible sound (incl. biomedical signal, ultra-sound) * Coding of metadata describing type and position of sensors and exact time * Coding of metadata describing measurement conditions * Metadata for annotation of content, e.g. reason for recording (diagnosis, background). |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Bandwidth is not an issue because there is no transmission of content, but storage capacity is important * Real-time processing necessary for recording. * Power shall be low because recording devices are battery powered. |

|  |
| --- |
| * 1. **UC9 Flexible Medical Data** |
| Target phase: Phase 1 |
| In order to monitor patients' well-being or baby’s intents any place, any time, certain algorithm models are needed for getting out results in a quickly but long-term way without assistance of doctors, so a large amount of data such as people's cough, breathing, sigh, snore, crying, sobbing, moaning, laughing, speech abnormality etc. shall be used as training data to train the algorithm models:   * Training data for mental health monitoring (e.g. stress levels, depressive disorder, anxiety disorder) * Training data for physical health early diagnosis (e.g. respiratory diseases, heart diseases) * Training data for sub-health analysis (e.g. stress levels, sleep disorders ) * Training data for baby’s intents (hungry, uncomfortable, want to play, diaper change) and states (awake, sick, sleeping) |
| **Overlap with other use cases** |
| None |
| **Required properties of the algorithm** |
| * Capable to encode multiple microphone channels to discard environmental noise. * Capable to encode metadata (see sub-tasks) * Capable to encode the configuration of multiple microphones in wearable and portable devices. * Capable to long-term monitoring but only record and diagnose when certain body sound event is triggered. * Optional privacy compliance: Automatic detection and suppression of speech recordings * All captured data must be time-stamped to track baby’s activities and sounds |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Metadata of the body sound characteristics: the frequency, intensity and duration (e.g. cough, snore, breathing) * Metadata for real-time body data (e.g. body temperature, heart rate) * Metadata for annotation of content, e.g. basic information and family medical history * Metadata for annotation of baby’s sounds (e.g. baby’s intents, emotions, states) * Metadata for annotation of baby’s movement sounds (e.g. baby’s intents, movement sounds description (clapping, kicking, scratching, turn over, eat fingers)). |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Compared to other use-cases, the required bandwidth shall be low because the connection might be wireless. * Real-time processing is necessary for recording. * Power shall be low because devices are battery powered. |

|  |
| --- |
| * 1. **UC10 UGC Analysis** |
| Target phase: Phase 1 |
| User Generated Content (UGC) is uploaded to platforms, which generates large amount of audio/video contents every day. Recommendation systems are deployed on servers to provide users with personalized content they are interested in. Audio embedding is one of those important features used by recommendation systems, usually, it is extracted by a neural network when audio content is uploaded to a platform. |
| **Overlap with other use cases** |
| UC11 Live Stream Content Analysis |
| **Required properties of the algorithm** |
| * Capable to encode multiple channels * Capable to encode metadata (see sub-tasks) |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing and near-lossless coding of all streams exploiting redundancies * Metadata for annotation of content, e.g., type of audio, activity of voice and music |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * The required bandwidth shall be low and storage capacity is also important. * Real-time processing is necessary. * Power is not a large concern. |

|  |
| --- |
| * 1. **UC11 Live Stream Content Analysis** |
| Target phase: Phase 1 |
| Live stream is generated by streamers with their microphones in a specific acoustical environment and consumed by users with their smart devices like smart phone, tablet or TV. Live stream content analysis is extracting useful information (e.g., is background sound or not, audio embedding) from live stream to improve the overall user experience. Neural networks are used to train detectors. The result of detector may change over time, because the audio content is also changing. |
| **Overlap with other use cases** |
| UC10 UGC Analysis |
| **Required properties of the algorithm** |
| * Capable to encode a few channels * Capable to encode metadata (see sub-tasks) * All captured data must be time-stamped to track audio labels. |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Capturing and near-lossless coding of all streams exploiting redundancies * Coding of metadata describing exact time * Metadata for annotation of content, e.g., type of audio, activity of voice and music. |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * The required bandwidth shall be low because audio content is uploaded and distributed via Internet * Real-time processing necessary. * Power shall be low because devices are battery powered. |

|  |
| --- |
| * 1. **UC12 Artistic creation** |
| Target phase: Phase 1 |
| Artistic creation provides corresponding audio contents, such as music composition, sound effect generation, removing ambient noise or removing human speech from the audio, isolating individual sounds in an auditory mixture of multiple sounds, source editing, etc. which provide assistance with audio processing from consumer to professional levels. |
| **Overlap with other use cases** |
| UC07 Speech Recognition and Acoustic Scene Analysis  UC10 UGC Analysis  UC11 Live Stream Content Analysis |
| **Required properties of the algorithm (Need further study)** |
| * Capable to encode multiple channels * Capable to encode decode computer script data * Capable to search part of audio data * Capable to decode part of audio data * Capable to decode metadata |
| **Optional properties of the algorithm** |
| * None |
| **What are the different sub-tasks expected in this use case?** |
| * Coding of audio script data, sub-title, dialogue, musical note * Coding of metadata describing environmental conditions (e.g. temperature, humidity, air pressure) * Metadata for annotation of contents (Music instrument playing information: music score, string bow movements) * Metadata for spec of recording equipments (model, mic numbers, position) |
| **What is the expected bandwidth of the distribution channel?**  **What is the maximum latency that is acceptable?**  **What are the power requirements?** |
| * Bandwidth is not an issue because there is no transmission of content * Real-time processing is necessary. * Power shall be low because devices are battery powered. |

# Summary of Proposed Sub-tasks (phase 1)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Description | UC1 | UC2 | UC3 | UC4 | UC5 | UC6 | UC7 | UC8 | UC9 | UC 10 | UC 11 | UC12 |
| Spatial Scene Capturing | Capturing the whole (acoustic) scene exploiting redundancy of recorded signals | X | X | X | X | X | X | X | X | X | - | X | X |
| Scene Analysis | Localization of sound objects in the scene | (X) | (X) | - | - | X | X | X | - | - | - | - | X |
| Metadata  Sensors | Coding of metadata describing type and position of sensors | X | X | X | X | X | X | X | X | X | - | - | X |
| Metadata environ-ment | Coding of metadata describing measurement conditions (temperature, humidity, air pressure) | X | X | X | X | X | X | X | X | X | - | - | X |
| Metadata process | Coding of processing steps in recorded data (“what was measured?”) | X | X | X | X | X | X | X | X | X | X | X | X |
| Metadata condition | Coding of operation conditions (OK/NOK) | X | X | X | X | - | - | - | X | - | - | - | - |
| Sound activity | Detection of sound activity | X | X | X | X | X | X | X | X | X | - | - | X |
| Sound content | Coding of sound context in recorded data | X | X | X | X | X | X | X | X | X | - | - | X |
| Speech activity | Detection of speech activity to suppress storage of speech | - | - | - | - | X | X | X | - | X | - | - | X |
| Speech content | Coding of spoken text in recorded data | - | - | - | - | - | - | X | - | - | - | - | - |
| Background Sound | Detection of background sound | - | - | - | - | - | - | - | - | - | - | X | X |
| Coding of timed data | (lossy) of scene with time stamp | X | - | - | - | X | X | - | X | X | - | X | X |

# Requirements

The term “machine” refers to a process or algorithm that gets as input audio or feature (eventually after a decoding stage) in order to analyse it or process it. For example, a machine is a neural network with the task to detect people in the input audio.

The potential benefits of ACoM include compression efficiency, computational offloading, and privacy protection, etc. The following requirements reflect different use-cases.

**Compression efficiency**

1. The specification shall offer a compression scheme which allows representing the essence of the data stream with a smaller size than traditional lossless audio coding.
2. The specification shall support varying levels of quality / coding efficiency as measured by performance for different sub tasks
   1. Some machines may be required to perform more accurately than others (i.e., the tasks that some machines perform may have higher priority or importance than the tasks performed by other machines).
   2. Level of quality may be a function of latency, bandwidth, or other application-specific requirements which may result in the varying encoding of the data stream.
3. The specification shall support near-lossless coding by exploiting redundancy in multi-channel data so that coding error must be lower than measurement precision.

**Coding bitstream**

1. The specification shall define a coding bitstream that can be usable and optimized for different scenarios;
   1. One bitstream for single task. Example: scene-level classification.
   2. One common bitstream for multiple tasks. Examples: object-level classification and action recognition.

**Privacy protection**

1. The specification shall support the option to guarantee privacy protection of the coded data.
   1. Note: This would not apply to reconstruction of spoken text and recognition of talker.

**Metadata coding**

1. The specification shall support the option to encode metadata about recording configuration and conditions, process recorded and status.
   1. For example, such metadata would include annotations of spoken text, audio label..

**Hybrid machine and human consumption**

1. The specification shall define a common coding bitstream which can be used for both machine and human consumption.

**Edge Computing and time stamps**

1. The specification shall support edge computing in decentral sensor nodes.
2. The specification shall define means to combine time aligned signals from various sensor nodes.

**Frequency bandwidth**

1. The specification shall support the coding of non-audible sounds for human.

**Number of Channels**

1. The specification shall support a varying number of audio channels.

# References

1. WG2, Technical Requirements, “Scope and Roadmap for Audio Coding for Machines (ACoM),” ISO/IEC JTC1/SC29/WG2 N0343, January 2024.
2. WG6, Technical Requirements, “Use Cases and Requirements on Audio Coding for Machines (ACoM),” ISO/IEC JTC1/SC29/WG6 N0252, April 2024.

**ANNEX**

1. **Other relevant groups**

Several other groups which are active in the application area have been identified.

* 1. **DCASE**

The DCASE Community (Detection and Classification of Acoustic Scenes and Events) is a scientific community where experts on algorithms work together. DCASE organizes workshops and challenges to compare algorithms based on real world data. The 2022 challenges included the following tasks:

1. Low-Complexity Acoustic Scene Classification
2. Unsupervised Anomalous Sound Detection for Machine Condition Monitoring Applying Domain Generalization Techniques
3. Sound Event Localization and Detection Evaluated in Real Spatial Sound Scenes
4. Sound Event Detection in Domestic Environments
5. Few-shot Bioacoustic Event Detection
6. Automated Audio Captioning and Language-Based Audio Retrieval

Tasks 2 and 3 of DCASE2022 are very close to the already identified use cases. Tasks 1, 4, 5 and 6 are use cases hadn’t been listed in document WG6/N0172.

An important comment from DCASE is that the topic “blind quality assessment” is still an open issue and needs further research before standardization on algorithms can start. This is in line with document WG6/N0172 which focuses on a lossless format. However, in future there might be a phase 2 of ACoM with extensions based on audio features adequate for algorithmic detection tasks providing lower data rates.

As mentioned above currently different formats for the audio data are used in this research community making exchange of data difficult. The DCASE community would benefit from a common format.   
Several members in WG6 are also active in DCASE.

* 1. **OPC**

The OPC Foundation (Open Platform Communications) is an industry consortium which creates and maintains standards for open connectivity of industrial automation devices and systems, such as industrial control systems and process control generally. The OPC standards specify the communication of industrial process data, alarms and events, historical data and batch process data between sensors, instruments, controllers, software systems, and notification devices.

Some experts reported that OPC enables them to optimize the whole manufacturing plant especially concerning fast reconfiguration, which is essential for cost efficient manufacturing of small batch sizes. To enable this each machine and each tool has a digital twin. That way production planning can model the complete production chain before the machines are reconfigured.

The set of standards in OPC currently does not include acoustical data.

* 1. **DICONDE**

Digital Imaging and Communication in Non-Destructive Evaluation (DICONDE) is an open standard format for the display, transfer and storage of digital non-destructive evaluation data. DICONDE belongs to ASTM International (formerly known as “American Society for Testing and Materials”) which is part of ISO. The list of method specific standards in DICONDE includes “Computed Radiography”, “Digital Radiography”, “Computed Tomography”, “Ultrasonic Test” and “Eddy Current Test”. DICONDE originally started as an extension of the well-established data format for medical image data DICOM.

The set of standards in DICONDE currently does not include acoustical data.

* 1. **DICOM**

The Digital Imaging and Communications in Medicine (DICOM) currently plans to extend their work to timed medical data (including electroencephalography (EEG), video-electroencephalography (VEEG), electromyography (EMG), evoked potentials (EP), polysomnograms (PSGs), electrocardiograms (ECGs), and other types of neurophysiology signals). First candidates to store such data are some MPEG audio formats.

* 1. **EDF**

The European Data Format (EDF) is widely used for the storage of multi-channel biomedical signals, supporting different sample rates per stream, annotations, etc. It is an open file format, with libraries for most common programming languages / data handling available.