Sevana Voice Quality Analyzer 3.4.10.327

Contents

Introduction

AQuA is a simple but powerful tool to provide perceptual voice quality testing and audio file comparison in terms of audio quality. This is the easiest way to compare two audio files and test voice quality loss between original and degraded files. Besides this a most demanded functionality of the software can also test audio codecs and generate audio signals for voice quality testing.

AQuA gives a unique opportunity to design your own voice quality testing solution not being dependent on particular hardware and software. It is available for Windows as a DLL and also available for Linux, portable to Java and mobile devices.

Functionality

Requirements

There are different versions of AQuA working with audio files represented in .wav or .pcm formats. Audio files should have the following characteristics depending on the version one uses

AQuA Voice: 8kHz, 16 bit, Mono.
AQuA HD Voice: 16kHz and up to 22.5kHz, 16 bit, Mono, Stereo
AQuA WB: 16kHz and up to 44.1kHz, 16 bit, Mono, Stereo

Generate test signals

Aqua allows generating test signals with the following parameters:

- Choosing voice type for synthesized sound: male or female
- Allows to define name of the synthesized sound or to run generation of full speech sounds distribution
- Allows choosing duration for synthesized sound equal in amount of samples
- Signals are built according to internal speech model
- Allows test sound signals generated as short, normal and long

Test voice codecs

- Allows output of codec speed performance indicator
- Allows testing any codec library

Compare wav files

- Allows intrusive testing of original wav file against degraded wav file
- Allows testing using internally generated audio test files
- Allows direct wav files comparison quality wise
- Allows measuring voice quality for any language

Testing parameters

AQuA supports the following parameters for voice quality testing:

- Choosing type of quality measurement: overall quality loss or voice naturalness
- Choosing filenames for original, test and generated audio files
- Define type of weight coefficients: uniform, linear or logarithmic
- Allows energy normalization
- Allows setting envelope smoothing level from 1 to 10
- Allows choosing source for original sound (external file or generated internally)
- Contains audio synchronization and voice activity detection
- Provides reasons for voice quality loss (quite unique feature on the market)
    - Duration distortion
    - Changes in signal spectrum
    - Distortion detection in low, medium and high frequency bands
- Provides voice quality feedback in:
    - Percentage of similarity
    - MOS-like value
    - PESQ-like value

Scientific background

The human ear is a non-linear system, which produces an effect named masking. Masking occurs on hearing a message against a noisy background or masking sounds.

As result of the research of the harmonic signal masking by narrow-band noise Zwiker has determined that the entire spectrum of audible frequencies could be divided into frequency groups or bands, recognizable by the human ear. Before Zwiker, Fletcher, who had named the selected frequency groups as critical bands of hearing, had drawn a similar conclusion.

Critical bands determined by Fletcher and Zwiker differ since the former has defined bands by means of masking with noise and the latter – from the relations of perceived loudness.

Sapozhkov has determined a critical band as “a band of frequency speech range, perceptible as a single whole”. In his earlier researches he even suggested that sound signals in a band could be substituted by an equivalent tone signal, but experiments did not confirm this assumption. Critical bands determined by Sapozhkov differ from those determined by Fletcher and Zwiker since Sapozhkov proceeded from the properties of speech signal.

Pokrovskij has also determined critical bands on the basis of speech signal properties. According to his definition the bands provide equal probability of finding formants in them.

The value of spectrum energy in bands can be used for different purposes; one of which is the sound signal quality estimation. However, using only one author’s critical bands (for example, Zwiker’s critical bands are used in prototype) does not allow getting an estimation objective enough, since they show only one of the aspects of perception or speech production. AQuA can determine energy in various critical bands as well as in logarithmic and resonator bands, that allows taking into consideration more properties of hearing and speech processing.

Taking into account that the bands determined by Pokrovskij and Sapozhkov are better for speech signal and not for sound signal, in general allows increasing the accuracy of estimation depending on it purpose.

AQuA utilized research results of the above mentioned scientists implementing different algorithms in one software solution. AQuA also has several advantages compared to other existing voice quality measurement software:

- it is universal since it allows measuring signals quality from various sources and processed in different ways;
- one can optimize quality estimation depending on the purposes:
    - for speed (for example, it is possible to receive rough estimation quickly);
    - in signal type (using different bands for speech signals and sound signals in general);

- resulting estimations correlate well with that of MOS;
- quality estimations received for speech signals can be translated in values of various kinds of intelligibility.

AQuA Command Line parameters

Print sounds’ names: -h sndn

sndn - prints list of sounds names;
There are 54 sounds in the database

```
-----------------------------------------------------------------------
Num   Name      Type                   Num   Name      Type
-----------------------------------------------------------------------
000    a0 <<--- Vocal                  001    a1 <<--- Vocal
002    a2 <<--- Vocal                  003    a4 <<--- Vocal
004    e0 <<--- Vocal                  005    e1 <<--- Vocal
006    i0 <<--- Vocal                  007    i1 <<--- Vocal
008    i4 <<--- Vocal                  009    o0 <<--- Vocal
010    o1 <<--- Vocal                  011    o4 <<--- Vocal
012    u0 <<--- Vocal                  013    u1 <<--- Vocal
014    u4 <<--- Vocal                  015    y0 <<--- Vocal
016    y1 <<--- Vocal                  017     l <<--- Sonor
018    l' <<--- Sonor                  019     m <<--- Sonor
020    m' <<--- Sonor                  021     n <<--- Sonor
022    n' <<--- Sonor                  023     j <<--- Sonor
024     v <<--- Noised                 025    v' <<--- Noised
026    zh <<--- Noised                 027     z <<--- Noised
028    z' <<--- Noised                 029     r <<--- Noised
030    r' <<--- Noised                 031     b <<--- Voiced Explosiv
032    b' <<--- Voiced Explosiv        033     g <<--- Voiced Explosiv
034    g' <<--- Voiced Explosiv        035     d <<--- Voiced Explosiv
036    d' <<--- Voiced Explosiv        037     f <<--- UnVoiced
038    f' <<--- UnVoiced               039     h <<--- UnVoiced
040    h' <<--- UnVoiced               041     s <<--- UnVoiced
042    s' <<--- UnVoiced               043    sh <<--- UnVoiced
044   sch <<--- UnVoiced               045     k <<--- Occlusive
046    k' <<--- Occlusive              047     p <<--- Occlusive
048    p' <<--- Occlusive              049     t <<--- Occlusive
050    t' <<--- Occlusive              051     c <<--- Occlusive
052    ch <<--- Occlusive              053     _ <<--- Silencer
-----------------------------------------------------------------------
```

Print samples of program usage: -h exam

exam - prints samples of program usage;

```
--------------------------------------------------------------
In order to test voice quality between original and reference files use the following set of parameters:

VQcmd.exe -mode files -src file ORIGINAL_FILE -tstf REFERENCE_FILE

To test voice codec provided as a DLL library use the following set of parameters:

VQcmd.exe -mode codec -clibf <DLL_LIBRARY_NAME> -src file <TEST_AUDIO_FILE>, e.g.
VQcmd.exe -mode codec -clibf GSM610.dll -src file short.wav
```

Define program mode> -mode <mod>

Defines AQuA mode of operation. The following modes are available:

<mod> :

codec - codec testing mode;
files - audio file comparison mode;
generate - test signals generation mode

For example: *-mode codec*

VQcmd.exe -mode files -src file ORIGINAL_FILE -tstf REFERENCE_FILE
VQcmd.exe -mode codec -clibf <DLL_LIBRARY_NAME> -src file <TEST_AUDIO_FILE>
VQcmd.exe -mode codec -clibf GSM610.dll -src file short.pcm

Command line argument: -clibf <file>

- codec library file name;

Use initial sound file as source or internal signal generator: -src <file <fname> | gen <mode>>

Determines source of initial sound: <file> - external sound file, or internal signal generator <gen>
In <file> mode one should specify name of audio file.
The signal generator mode has one of the following: short, normal or long

Set type of weight coefficients: -ct <ctype>

uniform, linear or logarithmic;

Set name of the file being tested: -tstf <fname>

Set name of the file generated by the speech model: -dst <fname>

Generate full speech sounds distribution or synthesized sound: -sn <all | <sname>>

runs generation of full speech sounds distribution or defines name of the synthesized sound

Examples:

VQcmd -sn all -dst SPEECH_MODEL_FILE

Here are options of generating **speech model** audio signal:
VQcmd.exe -mode generate -sn all -dst generated_01.pcm

instead of "all" parameter one can specify **separate sound**s from the table
of sounds you can see in the manual.
VQcmd.exe -mode generate -sn a0 -voit female -slen 8000 -dst
generated_02.pcm

VQcmd.exe -mode generate -sn i0 -voit male -slen 8000 -dst generated_03.pcm

For separate sounds one can also set type of voice "-voit male/female" and
duration of the sound to be generated "-slen 8000"

Setvoice type: -voit <female | male>

- sets voice type for synthesized sound;

Sets duration: -slen <num>

sets duration for synthesized sound equal to <num> samples

Set quality loss or naturallness: -qt <quality | naturalness>

- sets type of quality measurement **overall quality loss** or **voice naturalness**

Enable indication of codec speed performance: -power

enables output of codec speed performance indicator;

Enable energy normalization -enorm

- enables energy normalization;

Set number of link points: -npnt <num | auto>

sets number of link points;
auto - enables detection of optimal amount of linking points;

Set precision of spectral analysis: -acr <num | auto>

sets spectral analysis precision. num = 8..16,
auto - enables automated analysis precision detection according to sampling frequency.

Set envelope smoothing level: -miter <num>

smoothing level is in the range of (1..10);

Turn on "waiting for key press" after showing voice quality output: -gch

turns on waiting for a key press after output of voice quality

Print reason of quality loss: -fau <fname>

- prints reasons for quality loss to the file specified;

Set voice quality output type: -ratem <% | m | p>

%: voice quality in percentage,
M: MOS-like estimation
P: PESQ-like estimation.

AQuA Command Line usage

Most of our customers can be divided into two main categories:

- those who would like to compare original audio against degraded (test voice quality between the original and reference files) and obtain quantitative reasons for voice quality loss.
- those who like to compare two audio files and find out how similar they are quality wise

To show how this works we are going to use WAV files one can find on the site of Microtronix (http://www.microtronix.ca/pesq-disc.html). However, one can use any audio files that are recorded at 8Khz sampling and are 16 bit mono.

Compare two audio files and learn about reasons for voice quality loss

To compare two audio files in AQuA Command Line version when one is interested to get extensive feedback from the software we suggest to invoke AQuA in the following manner:

aqua-v.exe -mode files -src file Or272.wav -tstf Dg002.wav -acr auto -npnt auto -miter 1 -ratem %mp -fau log.txt

As result you will received the following output:

```
Sevana Audio Quality Analyzer - AQuA Voice v.2.1.0.151.
Copyright (c) 2009 by Sevana Oy, Finland. All rights reserved.
-------------------------------------------------------------
File Quality is
        Percent value   12.37
        MOS value        1.89
        PESQ value       1.72
```

Thus one can see that file comparison gives only 36.11% of similarity what corresponds to 1.49 MOS and about 2.56 PESQ. By the way, this is an example of when AQuA does detect voice quality loss and PESQ does not (please read more details about this test case on Microtronix page).

After test was exicute log.txt file contains quantitative reasons for voice quality loss:

```
Delay of audio signal activity.
Signal delayed by 120.000000 ms.

Audio signal activity mistiming (unsynchronization) is 1.73 percent.

Corrupted signal spectrum.
Overall spectral energy distortion approaches 87.07 %
Vibration along the whole spectrum [-22.79, 64.28] %

Significant distortion in low frequencies band.
Energy distortion approaches 35.36 %
Spectrum vibration in low frequency band [-19.46, 15.91] %

Significant distortion in medium frequencies band.
Energy distortion approaches 40.21 %
Amplification approaches 36.88 %
```

When invoking AQuA with the parameters above one receives the most detailed response from the software on voice quality and reasons for audio being degraded.

Compare two audio files and receive percentage of their similarity

In case we like to simply compare two audio files and get feedback on how similar they are quality wise we suggest invoking AQuA in the following manner:

```
aqua-v.exe -mode files -src file Or272.wav -tstf Dg001.wav  -acr auto -npnt auto -miter 1 -ratem %mp
```

Result will be:

```
Sevana Audio Quality Analyzer - AQuA Voice v.2.1.0.151.
Copyright (c) 2009 by Sevana Oy, Finland. All rights reserved.
-------------------------------------------------------------
File Quality is
        Percent value   79.46
        MOS value        4.28
        PESQ value       3.03
```

or invoking it for the other degraded file:

```
aqua-v.exe -mode files -src file Or272.wav -tstf Dg002.wav  -acr auto -npnt auto -miter 1 -ratem %mp
```

```
Sevana Audio Quality Analyzer - AQuA Voice v.2.1.0.151.
Copyright (c) 2009 by Sevana Oy, Finland. All rights reserved.
------------------------------------------------------------
File Quality is
        Percent value   12.37
        MOS value        1.89
        PESQ value       1.72
```

Exactly the same software parameters and testing approach are applicable to other versions of AQuA:

AQuA HD Voice – aqua-hdv.exe
AQuA Wideband – aqua-wb.exe

Tuning the algorithm for comparison of two audio files

This version of AQuA has default values for voice quality testing, set according to the majority of requests we have received from our customers. However, one can tune voice quality testing to better fit his own environment. In case your tests show unexpected results when comparing audio files this means that the AQuA engine or VAD may need tuning. We suggest to try playing around with these parameters first:

- -npnt
    - This parameter sets the amount of linking points required to catch different "holes" inside the signal. By default the value is 5.

- -miter
    - Sets amount of voice activity detector frames that are used during smoothing. By default it's 5. This is required to smooth the detector's vibration.

For example:

```
aqua-v.exe -mode files -src file Or272.wav  -tstf Dg002.wav -acr auto -npnt 1 -miter 5 -ratem %mp
```

Result is:

```
Sevana Audio Quality Analyzer - AQuA Voice v.2.1.0.151.
Copyright (c) 2009 by Sevana Oy, Finland. All rights reserved.
------------------------------------------------------------
File Quality is
        Percent value   12.91
        MOS value        1.89
        PESQ value       1.75
```

or invoking it for the other degraded file:

```
aqua-v.exe -mode files -src file Or272.wav  -tstf Dg001.wav  -acr auto -npnt 1 -miter 5 -ratem %mp -fau log.txt
```

```
Sevana Audio Quality Analyzer - AQuA Voice v.2.1.0.151.
Copyright (c) 2009 by Sevana Oy, Finland. All rights reserved.
------------------------------------------------------------
File Quality is
        Percent value   83.58
        MOS value        4.49
        PESQ value       3.10
```

In fact this result is much closer to what one would hear, however, the file was degraded.
One can find the reasons for voice quality loss in the log.txt file, e.g.:

Duration distortion.
Audio stretching corresponds to 11.72 percent.

Advancing of audio signal activity.
Signal advances the original by -350.000000 ms.

Audio signal activity mistiming (unsynchronization) is 7.07 percent.

Analysis of possible reasons for voice and audio quality loss

Besides audio quality score AQuA gives a possibility to analyze and determine possible reasons that caused audio signal degradation. Software automatically prepares analysis results that can be returned as a string or stored in a log file depending on the chosen option.

Additional audio quality metrics returned by the system may not look trivial to understand and this chapter is devoted to the main principles of how these metrics are built and how one can interpret them.

AQuA returns additional metrics only in the case when they are out of range for their "typical values". In case the metrics are within the range the system returns "Cannot determine the major reason for audio quality loss".

Duration distortion

This metric represents continuity of compared audio files. Ideally amount of audio data in the original signal and file under test should be the same. During audio processing or transfer over communication channels audio fragments may be lost as well as inserted into the audio. If such audio degradation took place then value of this metric is lower than 100. The bigger the difference the stronger the degradation, however, this metric does not consider possible starting pauses.

When the value is less than 100% this means that audio data was lost and analysis result will be:

Audio shrinking corresponds to XX.XX percent.

where XX.XX corresponds to deviation from 100%.

When the actual value is more than 100% this means that data was inserted and analysis result will be:

Audio stretching corresponds to XX.XX percent.

where XX.XX corresponds to deviation from 100%.

Tolerance range for this value is set to 100% ± 1%.

Delay/Advancing of audio signal activity

This metric represents signal shift in test file compared to the original and determines how much active level of the test signal delays/advances active level of the etalon (original) signal. When it is delayed analysis returns the following

Signal delayed by XX.XX ms.

where XX.XX is delay time in milliseconds. Correspondently, when the signal advances the original the return string is:

Signal advances the original by -XX.XX ms.

where XX.XX is advancing time.

Tolerance range for this value is interval of ±50 ms.

Audio signal activity mistiming

This metric represents unsynchronization of active levels in etalon and under test signals. Original (etalon) audio signal and test signal are merged to determine characteristics of audio activity, and when current characteristics of audio activity do not match system increases unsynchronization counter. After processing the final unsynchronization value is presented as percentage of cases when unsynchronization was detected.

If the metric value is not zero analysis result represent it as:

Audio signal activity mistiming (unsynchronization) is XX.XX percent.

where XX.XX is percentage of unsynchronization. The value is not considered if it is less than 1%.

Corrupted signal spectrum

This represents a set of metrics reflecting differences in integral energy spectrums of the original signal and audio under test. If overall spectrums difference is more than 15% than analysis returns the following string:

Corrupted signal spectrum.

If difference in spectrums is multidirectional (goes both into positive and negative zones) analysis returns the following string:

Vibration along the whole spectrum [-XX.XX, YY.YY] %

where XX.XX and YY.YY are deviations to negative and positive zones correspondently. Tolerance range of the deviation is ±5%.

If spectrum distortions are unidirectional (only negative or only positive) analysis returns this string:

Amplification approaches YY.YY %

When distortions are positive, or

Attenuation approaches XX.XX %

when distortions are negative.

Other metrics returned by analysis correspond to distortions occurred in different frequency groups. Analysis of different frequency bands performs in a similar manner to spectrum analysis. When talking about frequency bands in question we consider:

Low frequencies – below 1000 Hz

Medium frequencies – from 1000 Hz to 3000 Hz

High frequencies are those that are greater than 3000 Hz

When analyzing frequency bands we use other value tolerance ranges. Distortion in low frequencies is considered when they are greater than 5%, in medium frequencies – 10% and in high frequencies – 30%.

Multidirectional spectrum changes (vibration) is considered when they are greater than 2.5% in low frequencies, 7% in medium frequencies and 15% in high frequencies.

Unidirectional distortions (no matter positive or negative) are considered when they are greater than 5% in low frequencies, 10% in medium frequencies and 25% in high frequencies.

AQuA software benefits

Among AQuA benefits one will definitely appreciate that:

- AQuA is suitable to develop server solutions and does not have any limitations
- AQuA license does not have any annual royalty fee
- AQuA is suitable both for 32 and 64 bit systems
- AQuA is easy to deploy and use for software products development
- AQuA provides perceptual estimation of audio quality and can be utilized in VoIP, PSTN, ISDN, GSM, CDMA, LTE/4G networks and combinations of those