

ITU-T ITU-T P.863 Implementers' Guide

(05/2018)

SERIES P: Terminals and Subjective and Objective Assessment Methods

Methods for objective and subjective assessment of speech
quality

Implementers' Guide for Implementers' guide for P.863



Implementers' guide for P.863

Summary

This document contains all updates submitted up to and including those at Study Group 12 meeting in May 2018.

This document was approved by ITU-T Study Group 12 on 10 May 2018 and is the initial version of this implementers' guide.

Change log

2018-05

Initial version

Editors:

Jens Berger
Rohde & Schwarz, Switzerland
Joachim Pomy
OPTICOM GmbH, Germany

E-mail: jens.berger@rohde-schwarz.com

E-mail: consultant@joachimpomy.de

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications , information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

© ITU 2018

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

	Page
1. Scope.....	1
2. References.....	1
3. Introduction.....	1
4. Defect resolution procedure.....	1
5. Nomenclature.....	2
6. Technical and editorial corrections to Recommendation P.863.....	2
6.1. Annex B, TempAlignment.pdf.....	2
6.2. Annex B, TempAlignment.pdf.....	2
6.3. Annex B, TempAlignment.pdf.....	2
6.4. Annex B, TimeAlign.pdf.....	3
6.5. Annex B, SpeechTempAlign.pdf.....	4
6.6. Annex B, TempAlignment.pdf.....	4
6.7. Annex A, subfolder Results Ed. 3.....	4
6.8. Annex B - additional info.....	5
Annex A P.863 (2018-03) Defect Report Form.....	6

Implementers Guide ITU-T P.863 Implementers' Guide

Implementers' guide for P.863

1. Scope

This guide resolves defects in the following categories:

- editorial errors
- technical errors, such as omissions and inconsistencies
- ambiguities

In addition, the implementers' guide may include explanatory text found necessary as a result of interpretation difficulties apparent from the defect reports.

This guide will not address proposed additions, deletions or modifications to the Recommendations that are not strictly related to implementation difficulties in the above categories. Proposals for new features should be made through contributions to ITU-T.

2. References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T P.863 (03/2018)] Recommendation ITU-T P.863 (03/2018) (2018), *Perceptual objective listening quality prediction*, Third edition.

3. Introduction

This implementers' guide is a compilation of reported defects for the version 2018-03 of [\[ITU-T P.863 \(03/2018\)\]](#). In this edition of the guide, reported defects identified as of 2018-05 are given for:

P.863

The guide must be read in conjunction with Recommendation P.863 to serve as an additional source of information for implementers. The changes, clarifications and corrections defined herein are expected to be included in future versions of the affected Recommendations.

4. Defect resolution procedure

Upon discovering technical defects with any components of the texts covered by this implementers' guide, please provide a written description directly to the editors of the affected Recommendation(s) with a copy to the respective Rapporteur (See contacts above on page iii). The template for a defect report is located at the end of this guide. Return contact information should also be supplied so a dialogue can be established to resolve the matter and an appropriate reply to the defect report can be conveyed. This defect resolution process is open to any interested party. Formal membership in the ITU is not required to participate in this process.

5. Nomenclature

In addition to traditional revision marks, the following marks and symbols are used to indicate to the reader how changes to the text of a Recommendation should be applied:

Symbol	Description
<i>[Begin Correction]</i>	Identifies the start of revision marked text based on extractions from the published Recommendations affected by the correction being described.
<i>[End Correction]</i>	Identifies the end of revision marked text based on extractions from the published Recommendations affected by the correction being described.
...	Indicates that the portion of the Recommendation between the text appearing before and after this symbol has remained unaffected by the correction being described and has been omitted for brevity.
--- SPECIAL INSTRUCTIONS - -- {instructions}	Indicates a set of special editing instructions to be followed.

6. Technical and editorial corrections to Recommendation P.863

6.1. Annex B, TempAlignment.pdf

Missing statement, which rendered part of the surrounding code useless.

--- SPECIAL INSTRUCTIONS --- Insert at line 2655

[Begin Correction]

```
Done = true
```

[End Correction]

6.2. Annex B, TempAlignment.pdf

The fix from clause 6.1 requires setting larger search ranges for simple delay search problems. The old bounds were not used before due to the bug.

--- SPECIAL INSTRUCTIONS --- Replace lines 3940 and 3941:

[Begin Correction]

--- SPECIAL INSTRUCTIONS --- Old Description

```
mpReparsePoints[r].MinDelayVarInSamples = -2*WorstResolutionInSamples  
mpReparsePoints[r].MaxDelayVarInSamples = 2*WorstResolutionInSamples
```

--- SPECIAL INSTRUCTIONS --- New Description

```
mpReparsePoints[r].MinDelayVarInSamples = -4*WorstResolutionInSamples  
mpReparsePoints[r].MaxDelayVarInSamples = 4*WorstResolutionInSamples
```

[End Correction]

6.3. Annex B, TempAlignment.pdf

The fix from clause 6.1 revealed that before combining sections of constant delay, it was not sufficiently checked that both sections are either active speech or silence.

--- SPECIAL INSTRUCTIONS --- Modify lines 5092 to 5102:

[Begin Correction]

--- SPECIAL INSTRUCTIONS --- Old Description


```

for (utt=0 utt<NumberOfUtterances-1 utt++)
{
    if (pStartSampleUtterance[utt+1]-pStopSampleUtterance[utt] <
0.1*mProcessData.mSamplerate)
    {
        bool IsInsideActiveSection = true
        while (LastUsedReparsePoint<NumReparsePoints-1 &&
pStartSampleUtterance[utt+1]>ReparsePoints[LastUsedReparsePoint].Deg.End)
            LastUsedReparsePoint++
        if (pStartSampleUtterance[utt+1]<ReparsePoints[LastUsedReparsePoint]
.Deg.Start)
            IsInsideActiveSection = false

            if (!!IsInsideActiveSection &&

```

--- SPECIAL INSTRUCTIONS --- New Description

```

for (utt=0 utt<NumberOfUtterances-1 utt++)
{
    bool IsInsideActiveSection1 = true
    bool IsInsideActiveSection2 = true
    while (LastUsedReparsePoint<NumReparsePoints-1 &&
pStartSampleUtterance[utt]>ReparsePoints[LastUsedReparsePoint].Deg.End)
        LastUsedReparsePoint++
    if (pStartSampleUtterance[utt]<ReparsePoints[LastUsedReparsePoint].Deg.
Start && pStopSampleUtterance[utt]<ReparsePoints[LastUsedReparsePoint].Deg.
Start)
        IsInsideActiveSection1 = false

        int ReparsePoint2 = LastUsedReparsePoint
        while (ReparsePoint2<NumReparsePoints-1 && pStartSampleUtterance[utt+1]>
ReparsePoints[ReparsePoint2].Deg.End)
            ReparsePoint2++
        if (pStartSampleUtterance[utt+1]<ReparsePoints[LastUsedReparsePoint].
Deg.Start && pStopSampleUtterance[utt+1]<ReparsePoints[ReparsePoint2].Deg.Start)
            IsInsideActiveSection2 = false

            if (ReparsePoint2==LastUsedReparsePoint && (IsInsideActiveSection1 &&
IsInsideActiveSection2) || (!!IsInsideActiveSection1 && !IsInsideActiveSection2))
            {
                if (!!IsInsideActiveSection1 &&

```

[End Correction]

6.4. Annex B, TimeAlign.pdf

When building a histogram of correlations, the buffers of the correlations where not shifted, thus all values in the histogram were the same.

--- SPECIAL INSTRUCTIONS --- replace line 878

[Begin Correction]

--- SPECIAL INSTRUCTIONS --- Old Description

```
pDest[p] = matPearsonCorrelation(pA, pB+p, LenA)
```

--- SPECIAL INSTRUCTIONS --- New Description

```
pDest[p] = matPearsonCorrelation(pA+Offset, pB+p+Offset, LenA);
```

[End Correction]

6.5. Annex B, SpeechTempAlign.pdf

Avoid round-off errors by bounding the noise threshold at the lower end. This is only important to make porting to other platforms easier.

--- SPECIAL INSTRUCTIONS --- Insert at line 1439

[Begin Correction]

```
if (*NoiseLevelSilence<0.125) *NoiseLevelSilence = 0.125
```

[End Correction]

6.6. Annex B, TempAlignment.pdf

Avoid calculating correlation for signal segments which do not contain speech.

--- SPECIAL INSTRUCTIONS --- replace line 5146

[Begin Correction]

--- SPECIAL INSTRUCTIONS --- Old Description

```
int Delay = FindSectionAInSectionB(&SecA, &SecB, &RefSig, &DegSig, 0, 1, 1,
2*MaxLagSamples);
```

--- SPECIAL INSTRUCTIONS --- New Description

```
int FrameNumStart = floor((float) (SecB.Start)/(float)FrameSize);
int FrameNumEnd = floor((float) (SecB.End)/(float)FrameSize);
int FrameNum = FrameNumStart;
bool ActiveSeg = false;
while ((!ActiveSeg) && (FrameNum<=FrameNumEnd) && (FrameNum>=FrameNumStart) &&
(FrameNum<ActiveFramesLen))
{
    ActiveSeg = ActiveSeg || ActiveFrames[FrameNum];
    FrameNum++;
}
if (!ActiveSeg)
    continue;
int Delay = 0;
Delay = FindSectionAInSectionB(&SecA, &SecB, &RefSig, &DegSig, 0, 1, 1,
2*MaxLagSamples);
```

[End Correction]

6.7. Annex A, subfolder Results Ed. 3

The fixes in clauses 6.1 through 6.6 lead to minor changes in the conformance test results. Therefore the subfolder containing the old result files has to be replaced by the subfolder containing the new result files provided in the attached zip file.

--- SPECIAL INSTRUCTIONS --- Replace subfolder in Annex A

[Begin Correction]

--- SPECIAL INSTRUCTIONS --- Old Subfolder

Results Ed. 3

--- SPECIAL INSTRUCTIONS --- New Subfolder

Results Ed. 3 - IG 2018-05

[End Correction]

6.8. Annex B - additional info

NOTE – The fixes from clauses 6.1 through 6.6 have been applied to the Detailed Description files and a new set of all these pdf-files is provided as an electronic attachment.

Annex A

P.863 (2018-03) Defect Report Form

(This annex forms an integral part of this Implementers' Guide.)

DATE:	
CONTACT INFORMATION NAME: COMPANY: ADDRESS: TEL: FAX: E-MAIL:	
AFFECTED RECOMMENDATIONS:	
DESCRIPTION OF PROBLEM:	
SUGGESTIONS FOR RESOLUTION:	
NOTE – Attach additional pages if more space is required than is provided above.	