



НАЦИОНАЛЬНЫЙ
СТАНДАРТ
РОССИЙСКОЙ
ФЕДЕРАЦИИ

ГОСТ Р
55556.6 —
2013

**Звуковое вещание цифровое
Кодирование сигналов звукового вещания с
сокращением избыточности для передачи по
цифровым каналам связи.**

**ЧАСТЬ III
(MPEG-4 AUDIO)**

Интерфейс преобразования текста в речь (TTSI)

ISO/IEC 14496-3:2009
(NEQ)

Издание официальное



Москва
Стандартинформ
2014

Предисловие

1 РАЗРАБОТАН Санкт-Петербургским филиалом Центрального научно-исследовательского института Связи «Ленинградское отделение» (ФГУП ЛО ЦНИИС)

2 ВНЕСЕН Техническим комитетом по стандартизации № 480 «Связь»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 22 ноября 2013 г. № 1703-ст

4 Настоящий стандарт разработан с учетом основных нормативных положений международного стандарта ИСО/МЭК 14496-3:2009 «Информационные технологии. Кодирование аудиовизуальных объектов. Часть 3. Аудио» (ISO/IEC 14496-3:2009 Information technology - Coding of audio-visual objects - Part 3: Audio (NEQ))

5 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в ГОСТ Р 1.0 – 2012 (раздел 8). Информация об изменениях к настоящему стандарту публикуется в годовом (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок – в ежемесячно издаваемом информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования – на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (gost.ru)

© Стандартинформ, 2014

Настоящий стандарт не может быть воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Звуковое вещание цифровое

КОДИРОВАНИЕ СИГНАЛОВ ЗВУКОВОГО ВЕЩАНИЯ С СОКРАЩЕНИЕМ ИЗБЫТОЧНОСТИ ДЛЯ
ПЕРЕДАЧИ ПО ЦИФРОВЫМ КАНАЛАМ СВЯЗИ.
ЧАСТЬ III (MPEG-4 AUDIO)

Интерфейс преобразования текста в речь (TTSI)

Sound broadcasting digital.

Coding of signals of sound broadcasting with reduction of redundancy for transfer on digital communication channels.

A part III (MPEG-4 audio).

Text to speech interface (TTSI)

Дата введения — 2014—09—01

1 Область применения

Стандарт определяет кодированное представление преобразования текста в речь *MPEG-4 Audio (M-TTS)* и его декодер для синтеза речи высокого качества и для того, чтобы задействовать различные приложения.

Стандарт предназначен для приложения к функциональности *M-TTS*, такой как функциональность анимации лица (*FA*) и совместимость кинофильмов (*MP*) с кодированным потоком битов. Функциональности *M-TTS* включают возможность использования просодической информации, извлеченной из естественной речи. Функциональности также включают приложения в переговорное устройство для инструментов *FA* и устройство дублирования для кинофильмов, используя форму губ и вводимую информацию о тексте.

Технология синтеза преобразования текста в речь (*TTS*) становится довольно распространенным инструментом интерфейса и начинает играть важную роль в различных областях приложения мультимедиа. При использовании функциональности синтеза *TTS* легко могут быть составлены мультимедийные контенты с дикторским текстом, не записывая естественный звук речи. Кроме того, функциональность синтеза *TTS* с анимацией лица (*FA*) / кинофильма (*MP*) возможно сделала бы содержание контента более выразительным. Технология *TTS* может использоваться в качестве устройства речевого выхода для инструментов *FA* и для дублирования *MP* с информацией о форме губ.

В *MPEG-4* общие интерфейсы определяются для синтезатора *TTS* и для функциональной совместимости *FA/MP*. Функциональные возможности *M-TTS* можно рассматривать как надмножество стандартной платформы *TTS*. Синтезатор *TTS* может также использовать просодическую информацию естественной речи в дополнение к входному тексту и генерировать синтезированную речь гораздо более высокого качества. Формат потока битов интерфейса в высшей степени удобен для пользователя: если некоторые параметры просодической информации недоступны, пропущенные параметры генерируются, используя предварительно установленные правила. Функциональность *M-TTS*, таким образом, простирается от обычной функции синтеза *TTS* до кодирования естественной речи и областей его приложения, то есть, от простой функции синтеза *TTS* до функций для *FA* и *MP*.

2 Термины и определения

В настоящем стандарте применены термины с соответствующими определениями, используемые в ГОСТ Р 53556.0-2009.

3 Символы и сокращения

<i>F0</i>	основная частота (частота основного тона)
<i>DEMUX</i>	демультиплексор
<i>FA</i>	анимация лица
<i>FAP</i>	параметр анимации лица
<i>ID</i>	идентификатор
<i>IPA</i>	Международный фонетический алфавит
<i>MP</i>	кинофильм
<i>M-TTS</i>	<i>TTS MPEG-4 Audio</i>
<i>STOD</i>	повествователь историй по требованию
<i>TTS</i>	преобразование текста в речь

4 Синтаксис потока битов преобразования текста в речь *MPEG-4 Audio*

4.1 *TTSSpecificConfig MPEG-4 Audio*

```

TTSSpecificConfig () {
    TTS_Sequence ()
}

```

Таблица 1 - Синтаксис *TTS_Sequence ()*

Синтаксис	Количество битов	Мнемоника
<i>TTS_Sequence ()</i>		
{		
<i>TTS_Sequence_ID</i> ;	5	<i>uimsbf</i>
<i>Language_Code</i> ;	18	<i>uimsbf</i>
<i>Gender_Enable</i> ;	1	<i>bslbf</i>
<i>Age_Enable</i> ;	1	<i>bslbf</i>
<i>Speech_Rate_Enable</i> ;	1	<i>bslbf</i>
<i>Prosody_Enable</i> ;	1	<i>bslbf</i>
<i>Video_Enable</i> ;	1	<i>bslbf</i>
<i>Lip_Shape_Enable</i> ;	1	<i>bvslbf</i>
<i>Trick_Mode_Enable</i> ;	1	<i>bslbf</i>
}		

4.2 Полезная нагрузка преобразования текста в речь *MPEG-4 Audio*

```

AIPduPayload {
    TTS_Sentence ();
}

```

Таблица 2 — Синтаксис *TTS_Sentence ()*

Синтаксис	Количество битов	Мнемоника
<i>TTS_Sentence ()</i> {		
<i>TTS_Sentence_ID</i> ;	10	<i>uimsbf</i>
<i>Silence</i> ;	1	<i>bslbf</i>
if (<i>Silence</i>) {		
<i>SilenceDuration</i> ;	12	<i>uimsbf</i>
}		
else {		
if (<i>Gender_Enable</i>) {		
<i>Gender</i> ;	1	<i>bslbf</i>
}		
if (<i>Age_Enable</i>) {	3	<i>uimsbf</i>
<i>Age</i> ;		
}		
}		

Окончание таблицы 2

Синтаксис	Количество битов	Мнемоника
<i>if (!Video_Enable && Speech_Rate_Enable) {</i> <i>Speech_Rate;</i> <i>}</i>	4	<i>uimbsbf</i>
<i>Length_of_Text;</i> <i>for (j = 0; j < Length_of_Text; j++) {</i> <i>TTS_Text;</i> <i>}</i>	12	<i>uimbsbf</i>
<i>TTS_Text;</i> <i>}</i>	8	<i>bslbf</i>
<i>if (Prosody_Enable) {</i> <i>Dur_Enable;</i> <i>F0_Contour_Enable;</i> <i>Energy_Contour_Enable;</i> <i>Number_of_Phonemes;</i> <i>Phoneme_Symbols_Length;</i> <i>for (j = 0; j < Phoneme_Symbols_Length; j++) {</i> <i>Phoneme_Symbols;</i> <i>}</i> <i>for (j = 0; j < Number_of_Phonemes; j++) {</i> <i>if (Dur_Enable) {</i> <i>Dur_each_Phoneme;</i> <i>}</i> <i>if (F0_Contour_Enable) {</i> <i>Num_F0;</i> <i>for (k = 0; k < Num_F0; k++) {</i> <i>F0_Contour_each_Phoneme;</i> <i>F0_Contour_each_Phoneme_Time;</i> <i>}</i> <i>}</i> <i>if (Energy_Contour_Enable) {</i> <i>Energy_Contour_each_Phoneme;</i> <i>}</i> <i>}</i> <i>}</i>	1	<i>bslbf</i>
<i>Dur_Enable;</i>	1	<i>bslbf</i>
<i>F0_Contour_Enable;</i>	1	<i>bslbf</i>
<i>Energy_Contour_Enable;</i>	10	<i>uimbsbf</i>
<i>Number_of_Phonemes;</i>	13	<i>uimbsbf</i>
<i>Phoneme_Symbols_Length;</i>	8	<i>bslbf</i>
<i>for (j = 0; j < Phoneme_Symbols_Length; j++) {</i> <i>Phoneme_Symbols;</i> <i>}</i> <i>for (j = 0; j < Number_of_Phonemes; j++) {</i> <i>if (Dur_Enable) {</i> <i>Dur_each_Phoneme;</i> <i>}</i> <i>if (F0_Contour_Enable) {</i> <i>Num_F0;</i> <i>for (k = 0; k < Num_F0; k++) {</i> <i>F0_Contour_each_Phoneme;</i> <i>F0_Contour_each_Phoneme_Time;</i> <i>}</i> <i>}</i> <i>if (Energy_Contour_Enable) {</i> <i>Energy_Contour_each_Phoneme;</i> <i>}</i> <i>}</i> <i>}</i>	12	<i>uimbsbf</i>
<i>for (j = 0; j < Number_of_Phonemes; j++) {</i> <i>if (Dur_Enable) {</i> <i>Dur_each_Phoneme;</i> <i>}</i> <i>if (F0_Contour_Enable) {</i> <i>Num_F0;</i> <i>for (k = 0; k < Num_F0; k++) {</i> <i>F0_Contour_each_Phoneme;</i> <i>F0_Contour_each_Phoneme_Time;</i> <i>}</i> <i>}</i> <i>if (Energy_Contour_Enable) {</i> <i>Energy_Contour_each_Phoneme;</i> <i>}</i> <i>}</i> <i>}</i>	5	<i>uimbsbf</i>
<i>for (k = 0; k < Num_F0; k++) {</i> <i>F0_Contour_each_Phoneme;</i> <i>F0_Contour_each_Phoneme_Time;</i> <i>}</i> <i>}</i>	8	<i>uimbsbf</i>
<i>F0_Contour_each_Phoneme_Time;</i> <i>}</i>	12	<i>uimbsbf</i>
<i>if (Energy_Contour_Enable) {</i> <i>Energy_Contour_each_Phoneme;</i> <i>}</i> <i>}</i>	8*3=24	<i>uimbsbf</i>
<i>if (Video_Enable) {</i> <i>Sentence_Duration;</i> <i>Position_in_Sentence;</i> <i>Offset;</i> <i>}</i>	16	<i>uimbsbf</i>
<i>Sentence_Duration;</i>	16	<i>uimbsbf</i>
<i>Position_in_Sentence;</i>	10	<i>uimbsbf</i>
<i>Offset;</i> <i>}</i>		
<i>if (Lip_Shape_Enable) {</i> <i>Number_of_Lip_Shape;</i> <i>for (j = 0; j < Number_of_Lip_Shape; j++) {</i> <i>Lip_Shape_in_Sentence;</i> <i>Lip_Shape;</i> <i>}</i> <i>}</i>	10	<i>uimbsbf</i>
<i>Number_of_Lip_Shape;</i>	16	<i>uimbsbf</i>
<i>for (j = 0; j < Number_of_Lip_Shape; j++) {</i> <i>Lip_Shape_in_Sentence;</i> <i>Lip_Shape;</i> <i>}</i> <i>}</i>	8	<i>uimbsbf</i>
<i>Lip_Shape_in_Sentence;</i>		
<i>Lip_Shape;</i> <i>}</i> <i>}</i>		

5 Семантики потока битов преобразования текста в речь *MPEG-4 Audio*

5.1 *TTSSpecificConfig MPEG-4 Audio*

TTS_Sequence_ID – пятиразрядный ID, предназначенный однозначно определить каждый объект TTS, появляющийся в одной сцене. У каждого говорящего в сцене будет отличный *TTS_Sequence_ID*.

Language_Code – когда это "00" (00110000 00110000 в двоичном виде), IPA должен быть отправлен. В дополнение к этим 16 битам в конце добавляются два бита, которые представляют диалекты каждого языка (определяется пользователем).

Gender_Enable – однобитовый флаг, который устанавливается в '1', когда существует информация о половой принадлежности.

Age_Enable – однобитовый флаг, который устанавливается в '1', когда существует информация о возрасте.

Speech_Rate_Enable – однобитовый флаг, который устанавливается в '1', когда существует информация о темпе речи.

Prosody_Enable – однобитовый флаг, который устанавливается в '1', когда существует информация о просодии.

Video_Enable – однобитовый флаг, который устанавливается в '1', когда декодер *M-TTS* работает с *MP*. В этом случае *MTTS* должен синхронизировать синтетическую речь с *MP* и согласовать функциональность *ttsForward* и *ttsBackward*. Когда флаг *VideoEnable* устанавливается, *M-TTS* декодер использует системные часы, чтобы выбрать соответствующий фрейм *TTS_Sentence* и извлечь данные *Sentence_Duration*, *Position_in_Sentence*, *Offset*. Синтезатор TTS назначает подходящую продолжительность для каждой фонемы, чтобы обеспечить соответствие *Sentence_Duration*. Начальная точка речи в предложении определяется *Position_in_Sentence*. Если *Position_in_Sentence* равняется 0 (начальная точка является началом предложения), TTS использует *Offset* как время задержки, чтобы синхронизировать синтетическую речь с *MP*.

Lip_Shape_Enable – однобитовый флаг, который устанавливается в '1', когда кодированный входной поток битов содержит информацию о форме губ. При наличии информации о форме губ *M-TTS* просит инструмент *FA* изменить форму губ согласно информации о синхронизации (*Lip_Shape_in_Sentence*) и предопределяет конфигурацию формы губ.

Trick_Mode_Enable – однобитовый флаг, который устанавливается в '1', когда кодированный входной поток битов допускает такие специальные функции, как остановка, игра, движение вперед и назад.

5.2 Полезная нагрузка преобразования текста в речь *MPEG-4 Audio*

TTS_Sentence_ID – десятибитовый идентификатор, однозначно определяющий предложение в последовательности текстовых данных *M-TTS* для целей индексации. Первые пять битов равны *TTS_Sequence_ID* говорящего, а остальные пять битов являются последовательным номером предложения каждого объекта TTS.

Silence – однобитовый флаг, который устанавливается в '1', когда текущая позиция является молчанием.

Silence_Duration определяет продолжительность во времени текущего сегмента молчания в миллисекундах. Оно принимает значение от 1 до 4095. Значение '0' запрещается.

Gender – однобитовый флажок, который устанавливается в '1', если половая принадлежность производителя синтетической речи является мужской и '0', если женской.

Age представляет возраст говорящего для синтетической речи. Значение возраста определяется в таблице 3.

Таблица 3 — Таблица отображения возраста

Age	Возраст говорящего
000	менее 6
001	6 – 12
010	13 – 18
011	19 – 25
100	26 – 34
101	35 – 45
110	45 – 60
111	более 60

Speech_Rate - параметр определяет темп синтетической речи в 16 уровнях. Уровень 8 соответствует нормальному темпу речи говорящего, определенному в синтезаторе текущей речи, уровень 0 соответствует самой малой скорости синтезатора речи, а уровень 15 соответствует самой высокой скорости синтезатора речи.

Length_of_Text - параметр идентифицирует длину данных *TTS_Text* в байтах.

TTS_Text - строка символов, содержащая входной текст. Текст, заключенный в скобки < and >, содержит закладки. Если текст, заключенный в скобки < and >, начинается с *FAP*, закладка передается для анимации лица посредством *TtsFAPInterface* как строка символов. Иначе, текст закладки игнорируется.

Dur_Enable - однобитовый флаг, который устанавливается в '1', когда существует информация о продолжительности для каждой фонемы.

F0_Contour_Enable - однобитовый флаг, который устанавливается в '1', когда существует информация о контуре основного тона для каждой фонемы.

Energy_Contour_Enable - однобитовый флаг, который устанавливается в '1', когда существует информация о контуре энергии для каждой фонемы.

Number_of_Phonemes - параметр определяет число фонем, необходимых для синтеза речи из входного текста.

Phonemes_Symbols_Length - параметр идентифицирует длину данных *Phonemes_Symbols* (код *IPA*) в байтах, поскольку код *IPA* имеет коды дополнительных модификаторов и диалекта.

Phoneme_Symbols - параметр определяет номер индексации для текущей фонемы при использовании системы нумерации *Unicode 2.0*. Каждый символ фонемы представляется как число для соответствующего *IPA*. Для представления каждого *IPA* используются три двухбайтовых числа, включая двухбайтовое целое число для символа, и опционально двухбайтовое целое число для модификатора интервала, а также другое дополнительное двухбайтовое целое число для диакритического знака.

Dur_each_Phoneme - параметр определяет продолжительность каждой фонемы, мс.

Num_F0 - параметр определяет число значений *F0*, определенных для текущей фонемы.

F0_Contour_each_Phoneme - параметр определяет половину значения *F0*, Гц, в момент времени *F0_Contour_each_Phoneme_Time*.

F0_Contour_each_Phoneme_Time - параметр определяет целочисленное время, мс, для позиции *F0_Contour_each_Phoneme*.

Energy_Contour_each_Phoneme - три 8-битовых данных соответствуют значениям энергии в позиции старта, середины и окончания фонемы. Величина энергии *X* вычисляется как

$$X = \text{int} \left(50 \log_{10} A_{p,p} \right).$$

где $A_{p,p}$ является значением сигнала речи в размахе в определенной позиции.

Sentence_Duration - параметр определяет продолжительность предложения, мс.

Position_in_Sentence - параметр определяет позицию текущей остановки в предложении как прошедшее время, мс.

Offset - параметр определяет продолжительность очень короткой паузы перед стартом вывода синтезируемой речи, мс.

Number_of_Lip_Shape - параметр определяет число вариантов формы губ, которые будут обработаны.

Lip_Shape_in_Sentence - параметр определяет позицию каждой формы губ с начала предложения, мс.

Lip_Shape - параметр определяет число индексации для текущей реализации формы губ, которая будет обработана.

6 Процесс декодирования преобразования текста в речь *MPEG-4 Audio*

Предметом стандартизации архитектуры декодера *M-TTS* являются только интерфейсы, относящиеся к декодеру *M-TTS*.

В этой архитектуре различаются следующие типы интерфейсов:

интерфейс между демультимплексором и синтаксическим декодером;

интерфейс между синтаксическим декодером и синтезатором речи;

интерфейс от синтезатора речи к наборщику;

интерфейс от наборщика к синтезатору речи;

интерфейс между синтезатором речи и преобразователем фонем/закладок в *FAP*.

6.1 Интерфейс между демультимплексором и синтаксическим декодером

Получая поток битов, демультимплексор передает кодированные потоки битов *M-TTS* на

синтаксический декодер.

6.2 Интерфейс между синтаксическим декодером и синтезатором речи

Получая кодированный поток битов *M-TTS*, синтаксический декодер передает некоторые из следующих потоков битов на синтезатор речи.

Входной тип данных *M-TTS*: определяет синхронизированную работу с *FA* или *MP*

Поток команд управления: последовательность команд управления

Входной текст: строка(и) символов для текста, которая будет синтезирована

Вспомогательная информация: просодические параметры, включая символы фонем

Образцы формы губ

Информация для работы режима *trick*

Представление кода *pseudo-C* этого интерфейса.

6.3 Интерфейс от синтезатора речи к наборщику

Этот интерфейс идентичен интерфейсу для оцифрованной естественной речи в наборщике. Динамический диапазон от -32767 до $+32768$.

6.4 Интерфейс от наборщика к синтезатору речи

Этот интерфейс определяется, чтобы позволить локальное управление синтезируемой речи пользователями. Такой пользовательский интерфейс поддерживает режим приема синтезируемой речи в синхронизации с *MP* и изменяет некоторые просодические свойства синтезируемой речи путем использования *ttsControl*, определенного следующим образом:

Таблица 4 — Синтаксис *ttsControl()*

Синтаксис	Количество битов	Мнемоника
<pre> ttsControl() { ttsPlay(); ttsForward(); ttsBackward(); ttsStopSyllable(); ttsStopWord(); ttsStopPhrase(); TtsChangeSpeedRate(); TtsChangePitchDynamicRange(); TtsChangePitchHeight(); TtsChangeGender(); ttsChangeAge(); } </pre>		

Составляющая функция *ttsPlay* позволяет пользователю запускать синтез речи в прямом направлении, в то время как *ttsForward* и *ttsBackward* позволяют пользователю менять позицию запуска воспроизведения в прямом и обратном направлениях соответственно. Функции *ttsStopSyllable*, *ttsStopWord* и *ttsStopPhrase* определяют интерфейс для пользователей, чтобы останавливать синтез речи на указанной границе, такой как слог, слово и фраза. Составляющая функция *ttsChangeSpeechRate* является интерфейсом для изменения темпа синтезируемой речи. Параметр скорости принимает значения от 1 до 16. Составляющая функция *ttsChangePitchDynamicRange* является интерфейсом для изменения динамического диапазона основного тона синтезируемой речи. Используя параметр этой функции, уровень, пользователь может менять динамический диапазон от 1 до 16. Также пользователь может изменить высоту основного тона от 1 до 16 при использовании параметра высоты в составляющей функции *ttsChangePitchHeight*. Составляющие функции *ttsChangeGender* и *ttsChangeAge* позволяют пользователю изменять пол и возраст производителя синтетической речи, назначая значения их параметрам, полу и возрасту соответственно.

6.5 Интерфейс между синтезатором речи и конвертером фонем/закладок в *FAP*

В структуре *MPEG-4* синтезатор речи и анимация лица управляются синхронно. Синтезатор речи генерирует синтетическую речь. Одновременно *TTS* подает *phonemeSymbol* и *phonemeDuration*, а также закладки в конвертер *Phoneme/Bookmark-to-FAP*. Преобразователь фонем/закладок в *FAP* генерирует соответствующую анимацию лица согласно *phonemeSymbol*, *phonemeDuration* и закладкам.

Синтезируемая речь и анимация лица относительно синхронизированы кроме времени абсолютного смешивания. Синхронизация времени абсолютного смешивания приходит из той же самой отметки времени смешивания потока битов *TTS*. Если *Lip_Shape_Enable* устанавливается, то

Lip_Shape_in_Sentence используется, чтобы генерировать *phonemeDuration*. Иначе *TTS* обеспечивает продолжительности фонем. Синтезатор речи генерирует биты ударения и/или *wordBegin*, когда у соответствующей фонемы есть ударение, и/или начинается слово, соответственно.

В рамках *MTTS_Text* начало закладки для использования параметров анимации лица идентифицируется '<FAP'. Закладка длится до закрывающей угловой скобки '>'

Закладка подается *TtsFAPInterface* с фонемой следующего слова текущего предложения после закладки. Если после закладки нет никакого слова, закладка подается *TtsFAPInterface* с последней фонемой предыдущего слова в текущем предложении. Чтобы обеспечить анимацию сложных выражений и движения, разрешена последовательность до 40 закладок без слов между ними. *starttime* определяет время, мс, относительно начала последовательности *M-TTS*, когда фонема начнет воспроизводиться.

Класс *ttsFAPInterface* определяет структуру данных для интерфейса между синтезатором речи и конвертером *phoneme-to-FAP*.

Таблица 5 — Синтаксис *TtsFAPInterface* ()

Синтаксис	Количество битов	Мнемоника
<i>TtsFAPInterface</i> ()		
{		
<i>PhonemeSymbol</i> ;	8	<i>uimsbf</i>
<i>PhonemeDuration</i> ;	12	<i>uimsbf</i>
<i>f0Average</i> ;	8	<i>uimsbf</i>
<i>Напряжение</i> ;	1	<i>bslbf</i>
<i>WordBegin</i> ;	1	<i>bslbf</i>
<i>Закладка</i> ;		<i>char</i>
<i>Starttime</i> ;		<i>long int</i>
}		

Приложения декодера преобразования текста в речь MPEG-4 Audio

A.1 Общее

Эта часть приложения описывает прикладные сценарии для декодера *M-TTS*.

A.2 Прикладной сценарий: рассказчик истории MPEG-4 по требованию (STOD)

В приложении *STOD* пользователи могут выбрать историю из огромной базы данных библиотек истории, которые сохраняются на жестких дисках или компакт-дисках. Система *STOD* читает вслух историю через декодер *M-TTS* с инструментом анимации лица *MPEG-4* или с соответственно выбранными образами. Пользователь может остановить и продолжить воспроизведение в любой момент, когда он захочет, через пользовательские интерфейсы локальной машины (например, мышь или клавиатура). Пользователь может также выбрать пол, возраст, и темп речи электронного рассказчика историй.

Синхронизация между декодером *M-TTS* с инструментом анимации лица *MPEG-4* реализуется при использовании того же самого времени композиции декодера *M-TTS* для инструмента анимации лица *MPEG-4*.

A.3 Прикладной сценарий: преобразование текста в речь с кинофильмом MPEG-4 Audio

В этом приложении синхронизируемое воспроизведение декодера *M-TTS* и закодированного кинофильма является самой важной проблемой. Архитектура декодера *M-TTS* может обеспечить несколько степеней синхронизации. Выравнивая время смешивания каждого *TTS_Sentence*, может быть легко достигнута грубая степень синхронизации и функциональности режима приема. Чтобы получить более тонкую степень синхронизации, следует использовать информацию о *Lip_Shape*. Наиболее тонкая степень синхронизации может быть достигнута при использовании информации о просодии и связанной с видео информации, такой как *Sentence_Duration*, *Position_in_Sentence* и *Offset*.

С этой возможностью синхронизации декодер *M-TTS* может использоваться для копирования кинофильма, используя *Lip_Shape* и *Lip_Shape_in_Sentence*.

A.4 Закладки, использующие TTS и анимацию лица MPEG-4 Audio соответственно режиму спецэффектов

Закладки позволяют анимировать лицо, используя параметры анимации лица (*FAP*) в сочетании с анимацией рта, полученной из фонем. *FAP* закладки применяются к лицу, пока другая закладка не сбрасывает *FAP*. Разработка контентов, которые воспроизводят каждое предложение, независимое от режима спецэффектов, требует, чтобы закладки текста, которые будут произноситься, повторялись в начале каждого предложения, чтобы инициализировать лицо в состояние, которое определяется предыдущим предложением. В этом случае, может произойти некоторое несоответствие синхронизации в начале предложения. Однако система восстанавливается, когда обрабатывается новая закладка.

A.5 Модуль произвольного доступа

Каждое *TTS_Sentence* является модулем произвольного доступа.

Библиография

- [1] ИСО/МЭК 14496–3:2009 Информационные технологии. Кодирование аудиовизуальных объектов. Часть 3. Аудио (ИСО/МЭК14496–3:2009 *Information technology - Coding of audio-visual objects - Part 3: Audio*)

Ключевые слова: звуковое вещание, электрические параметры, каналы и тракты, технологии MPEG-кодирования, синтетический звук, масштабирование, защита от ошибок, поток битов расширения, психоакустическая модель

Подписано в печать 01.08.2014. Формат 60x84¹/₈.

Усл. печ. л. 1,40. Тираж 36 экз. Зак. 2862.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ФГУП «СТАНДАРТИНФОРМ»
123995 Москва, Гранатный пер., 4.
www.gostinfo.ru info@gostinfo.ru