

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G11B 20/10 (2006.01)



[12] 发明专利申请公开说明书

[21] 申请号 200480032931.4

[43] 公开日 2006年12月13日

[11] 公开号 CN 1879157A

[22] 申请日 2004.11.10

[21] 申请号 200480032931.4

[30] 优先权

[32] 2003.11.10 [33] KR [31] 10-2003-0079181

[32] 2004.10.19 [33] KR [31] 10-2004-0083517

[86] 国际申请 PCT/KR2004/002904 2004.11.10

[87] 国际公布 WO2005/045835 英 2005.5.19

[85] 进入国家阶段日期 2006.5.9

[71] 申请人 三星电子株式会社

地址 韩国京畿道

[72] 发明人 姜满锡 文诚辰 郑铉权

[74] 专利代理机构 北京铭硕知识产权代理有限公司

代理人 郭鸿禧 常桂珍

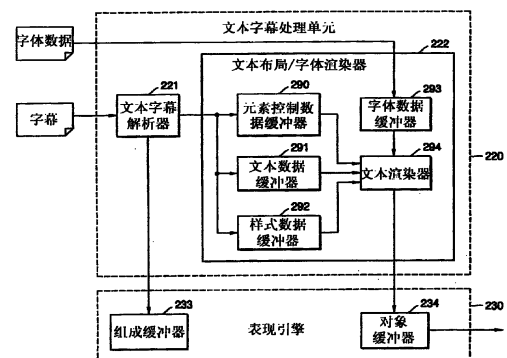
权利要求书 3 页 说明书 15 页 附图 11 页

[54] 发明名称

包含字幕的信息存储介质及其处理设备

[57] 摘要

一种包含字幕的信息存储介质和一种字幕处理设备，其中，所述信息存储介质包括：音视频(AV)数据；和字幕数据，在所述字幕数据中以文本格式存储至少一个字幕文本数据和用于指定所述至少一个字幕文本数据的输出形式的输出样式。从而，在文本字幕数据中包括的字幕文本的输出时间可能重叠，可容易地产生字幕文本，并且可用各种形式输出 AV 流的字幕。



1、一种信息存储介质，包括：

音视频数据；和

5 字幕数据，在所述字幕数据中以文本格式存储了至少一个字幕文本数据和用于指定所述至少一个字幕文本数据的输出形式的输出样式信息。

2、如权利要求1所述的信息存储介质，其中，所述输出样式信息包含多条信息以便输出样式信息被不同地应用于每个字幕文本。

10 3、如权利要求1所述的信息存储介质，其中，通过再现设备根据输出样式信息应用相同的输出样式并产生包括与所述字幕数据相应的一个图像的一页来渲染每个字幕文本数据。

4、如权利要求1所述的信息存储介质，其中，通过再现设备应用不同的输出样式并产生多页来渲染每个字幕文本数据，其中所述多页的每一页都包括每个相应的渲染的字幕文本数据。

15 5、如权利要求1所述的信息存储介质，其中，当存在多个字幕数据时，所述多个字幕数据被分别渲染为图像，渲染的图像分别包括多页。

6、如权利要求1所述的信息存储介质，其中，字幕数据还包括将所述至少一个字幕文本输出到屏幕上的时间的信息。

7、一种文本字幕处理设备，包括：

20 文本字幕解析器，从字幕数据分开地提取用于渲染文本字幕数据中的文本的渲染信息和用于表现渲染的文本的控制信息；和

文本布局/字体渲染器，根据提取的渲染信息渲染由文本字幕解析器提取的字幕文本来产生所述字幕文本的位图图像。

25 8、如权利要求7所述的设备，其中，所述文本字幕解析器提取控制信息以便所述控制信息适合于预定的信息结构格式，并将所述控制信息发送到表现引擎。

9、如权利要求7所述的设备，还包括：

30 文本字幕控制器，通过使用从用于处理位图字幕数据的表现引擎分离的控制信息来控制由文本布局/字体渲染器产生的位图图像被直接输出到屏幕上。

10、如权利要求7所述的设备，其中，所述字幕数据是具有重叠输出时

间的多个字幕数据。

11、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过产生与具有重叠输出时间的多个字幕数据相应的一个组成信息数据、一个位置信息数据和一个对象信息数据来产生位图，并输出该位图。

5 12、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过产生与具有重叠输出时间的多个字幕数据相应的一个组成信息数据、一个位置信息数据和多个对象信息数据来产生位图，并输出该位图。

13、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过产生与具有重叠输出时间的多个字幕数据相应的多个组成信息数据、一个位置信息数据和一个相应于每一组成信息数据的对象信息数据来产生位图，并输出该位图。

14、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过将相同的输出样式应用于多个文本字幕数据来渲染所述多个文本字幕数据的一个图像，并产生包括所述一个图像的一页。

15 15、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过将不同的输出样式应用于多个文本字幕数据的每一个来渲染所述多个文本字幕数据的图像，并产生包括所述多个渲染的图像的一页。

16、如权利要求7所述的设备，其中，所述文本布局/字体渲染器通过将不同的输出样式应用于多个文本字幕数据的每一个来渲染所述多个文本字幕数据的图像，并产生包括所述多个渲染的图像的多页。

17、一种使用信息存储介质的再现设备，包括：

读取器，从所述信息存储介质读取多个字幕，所述多个字幕的每个字幕都包括字幕文本、控制信息和样式信息；

缓冲器，对字幕、字幕文本、控制信息、样式信息缓冲；和

25 再现器，基于控制信息和样式信息对字幕解码，并根据控制信息和样式信息来显示字幕文本，其中，任意多个所述多个字幕都可被同时显示。

18、如权利要求17所述的设备，其中，所述再现器包括：

字幕处理器，从所述字幕的每一个提取字幕文本、控制信息和样式信息，并通过根据提取的控制信息和样式信息渲染提取的字幕来产生输出图像；和
30 表现引擎，显示所述输出图像。

19、如权利要求18所述的设备，其中，所述读取器读取非字幕位图图像，

所述表现引擎同时显示所述非字幕位图和所述输出图像。

20、如权利要求 17 所述的设备，其中，所述再现器包括：

字幕处理器，从所述字幕的每一个提取字幕文本、控制信息和样式信息，并根据提取的控制信息和样式信息渲染提取的字幕文本来产生输出图像；和
5 文本字幕控制器，根据提取的控制信息和样式信息来控制由所述字幕处理器产生的输出图像被显示在屏幕上。

21、一种再现文本字幕文件的方法，包括：

选择对于再现具有重叠的再现时间的字幕；
为所述字幕产生一组组合信息数据、位置信息数据和对象信息数据；和
10 根据组合信息数据、位置信息数据和对象信息数据的所述单一集合产生与字幕相应的图像。

22、一种再现文本字幕文件的方法，包括：

选择对于再现具有重叠的再现时间的字幕；
为所选择的字幕的每一个产生一组组合信息数据、位置信息数据和对象
15 信息数据，其中，所选择的每一个字幕的位置信息数据、对象信息数据互不相同；和

根据组合信息数据、对于每个字幕不同的位置信息数据和对于每个字幕不同的对象信息数据的单一集合产生与字幕相应的图像。

23、一种再现文本字幕文件的方法，包括：

20 选择对于再现具有重叠的再现时间的字幕；
为每一个字幕产生互不相同的一组组合信息数据、位置信息数据和对象信息数据的集合；和

根据对于每个字幕不同的组合信息数据、位置信息数据和对象信息数据的集合产生与字幕相应的图像。

25

包含字幕的信息存储介质及其处理设备

5 技术领域

本发明涉及一种信息存储介质，更具体地讲，涉及一种包含可被分开显示的多个字幕的信息存储介质及其处理设备。

背景技术

10 传统的字幕是包括在音视频(AV)流中的位图图像。因此，不方便制作这类字幕，并且由于用户不能够选择由字幕制作者定义的字幕的各种属性，所以没有选择只能不作修改以字幕目前的形式来读取该字幕。即，由于诸如字体、字符大小和字符颜色的属性被预定义并作为位图图像被包括在 AV 流中，所以用户不能够随意改变所述属性。

15 另外，由于在 AV 流中字幕被压缩并被编码，所以字幕的输出开始时间和输出结束时间被清楚地指定以与 AV 流相应，并且输出字幕的再现时间不应该重叠。即，在特定时间应该仅输出一种字幕。

然而，由于字幕的输出开始时间和输出结束时间由字幕制作者指定并与 AV 流分开记录在信息存储介质上，所以多个字幕的输出开始时间和输出结束
20 时间可能互相重叠。换句话说，由于在特定时间段内可能输出多于两种的字幕，所以解决这个问题的方法是必要的。

发明内容

技术方案

25 在本发明的一方面中，本发明提供了一种将尽管互相重叠但分开显示的多个文本字幕记录在其上的信息存储介质，以及一种再现该信息存储介质的设备。

有益的效果

30 根据本发明的实施例，可容易地制作字幕文件，并能够以多种形式输出用于 AV 流的字幕。

附图说明

图 1 示出文本字幕文件的结构;

图 2 是再现其上记录有文本字幕的信息存储介质的设备的方框图;

5 图 3 是图 2 的文本字幕处理单元的详细方框图;

图 4 是示出不利用表现引擎而产生位图的参考方框图;

图 5 是示出组成信息、位置信息、对象信息和颜色信息记录在其中的结构之间的相互关系的示图;

10 图 6A 至 6C 是示出通过使用一个组成信息数据和一个位置信息数据产生用于多个字幕的图像的过程的示图;

图 7A 至 7C 是示出通过使用一个组成信息数据和多个位置信息数据产生用于多个字幕的图像的过程的示图; 和

图 8A 至 8C 是示出通过以下方式产生图像的过程的示图, 即通过对多个字幕分配多个组成信息数据来将一个图像对象包括在一个组成信息数据中;

15

最佳方式

根据本发明的一方面, 提供了一种信息存储介质, 该信息存储介质包括: AV 数据; 和字幕数据, 在该字幕数据中以文本格式存储至少一个字幕文本数据或用于指定字幕文本的输出形式的输出样式信息。

20 在本发明的一方面, 输出样式信息包含多条信息, 以便输出样式信息被不同地应用于字幕文本。

在本发明的一方面, 当存在多个字幕数据时, 所述多个字幕数据被分开渲染, 渲染的图像分别包括多个页。

25 根据本发明的另一方面, 提供了一种文本字幕处理设备, 该文本字幕处理设备包括: 文本字幕解析器, 分别从文本字幕数据提取用于渲染文本字幕数据中的文本的渲染信息和用于表现渲染的文本的控制信息; 和文本布局/字体渲染器, 通过根据提取的渲染信息渲染字幕文本来产生字幕文本的位图图像。

30 在本发明的一方面, 文本布局/字体渲染器通过将不同样式应用于字幕文本数据来渲染至少一个字幕文本数据, 并组成具有多个渲染的图像的多页。

具体实施方式

现在将详细描述本发明的实施例，其示例将在附图中示出，其中，相同的标号始终表示相同的部件。以下将参照附图来描述所述实施例以解释本发明。

5 图 1 示出文本字幕文件 100 的结构。参照图 1，文本字幕文件 100 包括对话信息 110、表现信息 120 和元数据 130a 和 130b。

对话信息 110 包括字幕文本、字幕文本的输出开始时间、字幕文本的输出结束时间、用于渲染字幕文本的样式组或样式信息、诸如淡入和淡出的文本改变效果信息、和字幕文本的格式代码。格式代码包括一个或多个用粗体
10 字符显示文本的代码、用斜体显示文本的代码、指示下划线的代码或指示换行的代码。

表现信息 120 包括用于渲染字幕文本的样式信息，并包括多个样式组。样式组是一簇样式，在其上记录样式信息。所述样式包括用于渲染和显示字幕文本的信息。该信息包括，例如，一个或多个样式名、字体、文本颜色、
15 背景颜色、文本大小、行高、文本输出区域、文本输出开始位置、输出方向或对齐方法。

作为运动画面的附加信息的元数据 130a 和 130b 包括用于执行除了字幕输出功能之外的附加功能所需的信息。例如，所述附加功能可以是对于针对成年观众的节目在屏幕上显示诸如‘TV-MA’的 TV 家长指导。

20 图 2 是再现信息存储介质的设备的方框图，在该信息存储介质上记录文本字幕文件。应该理解，该设备还可将文本字幕文件记录到该信息存储介质。

参照图 2，文本字幕处理单元 220 渲染字幕文本以处理文本字幕文件。文本字幕处理单元 220 包括：文本字幕解析器 221，从文本字幕文件提取表现信息和对话信息；和文本布局/字体渲染器 222，根据提取的表现信息渲染
25 字幕文本来产生输出图像。

图 1 中示出的文本字幕文件 100 可被记录在再现设备包括的存储器中或信息存储介质上。在图 2 中，其上记录有文本字幕的信息存储介质或存储器被称为字幕信息存储单元 200。

从字幕信息存储单元 200 读取与正在再现的运动画面相对应的文本字幕
30 文件以及用于渲染所述字幕的字体数据，并将其存储在缓冲器 210 中。将缓冲器 210 中存储的文本字幕文件发送到文本字幕解析器 221，该文本字幕解

析器 221 对渲染文本字幕文件所需的信息进行解析。将字幕文本、字体信息和渲染样式信息发送到文本布局/字体渲染器 222，将字幕文本的控制信息发送到表现引擎 230 的组成缓冲器(composition buffer)233。控制信息(即，用于显示具有字幕文本的屏幕的信息)包括输出区域和输出开始位置。

- 5 文本布局/字体渲染器 222 通过使用从文本字幕解析器 221 发送的文本渲染信息和从缓冲器 210 发送的字体数据渲染字幕文本来产生位图图像，通过指定每个字幕文本的输出开始时间和输出结束时间来组成一个字幕页，并将位图图像和字幕页发送到表现引擎 230 的对象缓冲器 234。

从字幕信息存储单元 200 读取的位图图像形式的字幕被输入到编码的数
10 据缓冲器 231，并由表现引擎 230 中的图形处理单元 232 来处理。因此，图形处理单元 232 产生位图图像。将产生的位图图像发送到对象缓冲器 234，将该位图图像的控制信息发送到组成缓冲器 233。控制信息用于指定对象缓冲器 234 中存储的位图图像被输出到图形面板器(graphic planer)240 的时间和位置，并用于指定颜色查找表(CLUT)250，在该颜色查找表 250 中记录将被
15 应用于输出到图形面板器 240 的位图图像的颜色信息。组成缓冲器 233 接收从文本字幕解析器 221 发送的对象组成信息和由图形处理单元 232 处理的位图字幕数据，并将用于将字幕输出到屏幕上的控制信息发送到图形控制器 235。图形控制器 235 控制对象缓冲器 234 以将由图形处理单元 232 处理的位图字幕数据和从文本布局/字体渲染器 222 接收的渲染的文本对象数据组合，
20 并控制图形面板器 240 根据组合的数据产生图形面板，图形控制器 235 参照 CLUT 250 将该图形面板输出到显示单元(未示出)。

图 3 是图 2 的文本字幕处理单元 220 的详细方框图。参照图 3，将作为文本字幕文件信息的字幕输入到文本字幕解析器 221。文本字幕解析器 221 将从字幕解析的字幕控制信息发送到表现引擎 230，并将从字幕解析的文本
25 渲染信息发送到文本布局/字体渲染器 222。文本布局/字体渲染器 222 从文本字幕解析器 221 接收文本渲染信息，将字幕文本的控制信息存储在元素控制数据缓冲器 290 中，将字幕文本数据存储在文本数据缓冲器 291 中，将用于渲染字幕文本数据的样式信息存储在样式数据缓冲器 292 中。另外，文本布局/字体渲染器 222 将用于文本渲染的字体数据存储在字幕数据缓冲器 293
30 中。

元素控制数据缓冲器 290 中存储的控制信息可以是格式代码。格式代码

包括一个或多个用粗体字符显示文本的代码、用斜体显示文本的代码、指示下划线的代码或指示换行的代码。文本数据缓冲器 291 中存储的字幕文本数据是将被输出为字幕的文本数据。样式数据缓冲器 292 中存储的样式数据可以是一个或多个数据，例如字体、文本颜色、背景颜色、文本大小、行高、

5 文本输出区域、文本输出开始位置、输出方向或对齐方法。文本渲染器 294 参照记录在每个缓冲器中的信息产生字幕图像，并将该字幕图像发送到表现引擎 230。

图 4 是示出不利用表现引擎 230 而产生位图图像的参考方框图。

10 即，图 4 示出包括文本字幕控制器 410 而不是表现引擎 230 的文本字幕处理单元 220 的操作的另一实施例。

参照图 4，文本布局/字体渲染器 222 产生组成信息、位置信息、对象信息和颜色信息，并在组成信息、位置信息、对象信息和颜色信息的基础上产生位图图像。文本字幕控制器 410 从文本字幕解析器 221 接收对象组成信息，并控制文本布局/字体渲染器 222 将由文本布局/字体渲染器 222 产生的位图图

15 像直接输出到图形面板器 240 和 CLUT 250。

图 5 是示出其中记录有组成信息、位置信息、对象信息和颜色信息的结构之间的相互关系的示图。

被输出到屏幕上的字幕由多个页单元组成。每页还可包括用于除字幕之外的其他目的的数据。组成信息表示包含用于组成一页的信息的信息。组成

20 信息包括指示页输出时间的输出时间信息、指示输出图像对象的对象信息参考值、指示对象输出位置的位置信息参考值和指示对象颜色信息的颜色信息参考值。

图 5 中显示的信息结构之间的相互关系是组成信息的一部分，并且还可以以与图 5 不同的形式来组成位置信息、对象信息和颜色信息之间的相互关

25 系。

参照图 5，一页可包括用于将图像输出到屏幕上的至少一个区域。该至少一个区域根据位置信息参考值来分类。位置信息表示记录的组成所述至少一个区域以输出图像所需的信息的结构。位置信息包括每个区域的水平和垂直坐标、区域的宽度和区域的高度的信息。对象信息包括将被显示在屏幕上的

30 的对象数据。另外，对象信息包括与对象数据相应的对象数据类型信息。

将作为示例来描述文本字幕处理单元 220 的操作。文本字幕处理单元 220

产生将被输出到屏幕上的每个渲染的字幕图像的组成信息、位置信息、对象信息和颜色信息以提供字幕文本。将产生的组成信息、位置信息、对象信息和颜色信息发送到表现引擎 230。

5 如上所述，当包含以文本形式产生的字幕的信息存储介质被再现时，存在同时输出多于一个字幕的多种示例性方法。

在第一种方法中，文本字幕处理单元 220 产生用于文本输出时间重叠的多个字幕的新的图像，并将由对象组成的字幕发送到表现引擎 230，其中，产生的对象将被输出到一个组成信息中的一个位置信息。

10 存在组成字幕的第二种方法以便文本输出时间重叠的字幕具有不同的位置信息。即，文本字幕处理单元 220 通过使用一个组成信息中的不同的位置信息来产生文本输出时间重叠的多个字幕的图像，并将产生的图像发送到表现引擎 230。

15 存在通过使用不同的组成信息来产生文本输出时间重叠的字幕的第三种方法。即，文本字幕处理单元 220 对文本输出时间重叠的多个字幕产生不同的组成信息，以便仅一个对象被包括在一个组成信息数据中。

将参照图 6A 至 8C 来详细描述这三种方法。

图 6A 至 6C 是示出通过使用一个组成信息数据和一个位置信息数据来为多个字幕产生图像的过程示图。

20 在图 6A 中，样式‘Script’被定义为用于字幕文本渲染的样式信息。参照图 6A，样式‘Script’使用字体‘Arial.ttf’、文本颜色‘black’、背景颜色‘white’、字符大小‘16pt’、文本参考位置的坐标(x,y)，对齐方法‘center’、输出方向‘left-to-right-top-to-bottom’、文本输出区域‘left, top, width, height’和行高‘40px’。

25 在图 6B 中，定义通过使用样式‘Script’渲染的字幕文本 610、620 和 630。参照图 6B，从‘00:10:00’到‘00:15:00’输出字幕文本 Hello 610，从‘00:12:00’到‘00:17:00’输出字幕文本 Subtitle 620，从‘00:14:00’到‘00:19:00’输出字幕文本 World 630。因此，在‘00:12:00’和‘00:17:00’之间输出两个或三个字幕文本。这里，‘
’指示换行。尽管使用一个样式，可通过使用
来防止多个字幕在一个区域上重叠。

30 图 6C 显示输出在图 6A 和 6B 中定义的字幕的结果。参照图 6C，将详细地描述在每个示出的时间窗口中文本字幕处理单元 220 的每个缓冲器中存储

的数据。

在‘00:10:00’之前：当输出的组成信息包括空字幕图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：空；

5 文本数据缓冲器：空；

样式数据缓冲器：样式信息‘Script’；和

字体数据缓冲器：字体信息‘Arial.ttf’。

从‘00:10:00’到‘00:12:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 610 的图像时，文本布局/字体渲染器 222 包括：

10 元素控制数据缓冲器：字幕文本 Hello 610 的控制信息；

文本数据缓冲器：‘Hello’；

样式数据缓冲器：样式信息‘Script’；和

字体数据缓冲器：字体信息‘Arial.ttf’。

15 从‘00:12:00’到‘00:14:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 610 和字幕文本 Subtitle 620 的图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：字幕文本 Hello 610 和字幕文本 Subtitle 620 的控制信息；

文本数据缓冲器：‘Hello’和‘
Subtitle’；

样式数据缓冲器：样式信息‘Script’；和

20 字体数据缓冲器：字体信息‘Arial.ttf’。

从‘00:14:00’到‘00:15:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 610、字幕文本 Subtitle 620 和字幕文本 World 630 的图像时，文本布局/字体渲染器 222 包括：

25 元素控制数据缓冲器：字幕文本 Hello 610、字幕文本 Subtitle 620 和字幕文本 World 630 的控制信息；

文本数据缓冲器：‘Hello’、‘
Subtitle’和‘

World’；

样式数据缓冲器：样式信息‘Script’；和

字体数据缓冲器：字体信息‘Arial.ttf’。

30 从‘00:15:00’到‘00:17:00’：当输出的组成信息包括在其中渲染了字幕文本 Subtitle 620 和字幕文本 World 630 的图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：字幕文本 Subtitle 620 和字幕文本 World 630 的控

制信息;

文本数据缓冲器: ‘
Subtitle’和‘

World’;

样式数据缓冲器: 样式信息‘Script’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

- 5 从‘00:17:00’到‘00:19:00’: 当输出的组成信息包括在其中渲染了字幕文本 World 630 的图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 字幕文本 World 630 的控制信息;

文本数据缓冲器: ‘

World’;

样式数据缓冲器: 样式信息‘Script’; 和

- 10 字体数据缓冲器: 字体信息‘Arial.ttf’。

在‘00:19:00’之后: 当输出的组成信息包括空字幕图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 空;

文本数据缓冲器: 空;

- 15 样式数据缓冲器: 样式信息‘Script’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

如上所示的字幕输出过程, 在第一种方法中, 通过将相同样式应用到具有重叠输出时间的多个字幕文本来产生一个字幕图像, 产生包括所述一个字幕图像的一个组成信息数据, 并将产生的组成信息数据发送到表现引擎 230。

- 20 此时, 用于指示发送的组成信息从屏幕消失的时间的 $page_time_out$ 表示在具有重叠的输出时间的多个字幕中最后输出到屏幕的字幕的消失时间, 或者添加新的字幕的时间。

- 25 考虑到在文本字幕处理单元 220 中对字幕执行解码所花费的时间 $T_{decoding}$ 和将渲染的字幕从对象缓冲器 234 输出到图形面板器 240 所花费的时间 $T_{composition}$, 必须快速执行对输出字幕的文本字幕处理。当 T_{start} 指示从再现设备的文本字幕处理单元 220 输出字幕的时间时, 以及当 $T_{arrival}$ 指示字幕到达文本字幕处理单元 220 的时间时, 将通过方程 1 来计算这些时间之间的关系。

从‘00:12:00’到‘00:17:00’输出，字幕文本 World 730 使用样式‘Script3’，并从‘00:14:00’到‘00:19:00’输出。因此，在‘00:12:00’和‘00:17:00’之间输出了两个或三个字幕文本。由于使用了不同的脚本(script)，所以换行标签
是不必要的。

- 5 图 7C 显示输出在图 7A 和 7B 中定义的字幕的结果。参照图 7C，将详细描述每个示出的时间窗口中文本字幕处理单元 220 的每个缓冲器中存储的数据。

在‘00:10:00’之前：当输出的组成信息包括空字幕图像时，文本布局/字体渲染器 222 包括：

- 10 元素控制数据缓冲器：空；
 文本数据缓冲器：空；
 样式数据缓冲器：空；和
 字体数据缓冲器：字体信息‘Arial.ttf’。

- 15 从‘00:10:00’到‘00:12:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 710 的图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：字幕文本 Hello 710 的控制信息；
 文本数据缓冲器：‘Hello’；
 样式数据缓冲器：样式信息‘Script1’；和
 字体数据缓冲器：字体信息‘Arial.ttf’。

- 20 从‘00:12:00’到‘00:14:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 710 和字幕文本 Subtitle 720 的图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：字幕文本 Hello 710 和字幕文本 Subtitle 720 的控制信息；

- 25 文本数据缓冲器：‘Hello’和‘Subtitle’；
 样式数据缓冲器：样式信息‘Script1’和‘Script2’；和
 字体数据缓冲器：字体信息‘Arial.ttf’。

从‘00:14:00’到‘00:15:00’：当输出的组成信息包括在其中渲染了字幕文本 Hello 710、字幕文本 Subtitle 720 和字幕文本 World 730 的图像时，文本布局/字体渲染器 222 包括：

- 30 元素控制数据缓冲器：字幕文本 Hello 710、字幕文本 Subtitle 720 和字幕文本 World 730 的控制信息；

文本数据缓冲器: ‘Hello’、‘Subtitle’和‘World’;

样式数据缓冲器: 样式信息‘Script1’、‘Script2’和‘Script3’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

- 5 从‘00:15:00’到‘00:17:00’: 当输出的组成信息包括在其中渲染了字幕文本 Subtitle 720 和字幕文本 World 730 的图像时, 文本布局/字体渲染器 222 包括:
元素控制数据缓冲器: 字幕文本 Subtitle 720 和字幕文本 World 730 的控制信息;

文本数据缓冲器: ‘Subtitle’和‘World’;

样式数据缓冲器: 样式信息‘Script2’和‘Script3’; 和

- 10 字体数据缓冲器: 字体信息‘Arial.ttf’。

从‘00:17:00’到‘00:19:00’: 当输出的组成信息包括在其中渲染了字幕文本 World 730 的图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 字幕文本 World 730 的控制信息;

文本数据缓冲器: ‘World’;

- 15 样式数据缓冲器: 样式信息‘Script3’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

在‘00:19:00’之后: 当输出的组成信息包括空字幕图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 空;

- 20 文本数据缓冲器: 空;

样式数据缓冲器: 空; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

- 在上述第二种方法中, 通过将不同的样式应用到具有重叠输出时间的多个字幕文本来为字幕文本产生字幕图像, 产生包括所述字幕图像的一个组成
25 信息数据, 并将产生的组成信息数据发送到表现引擎 230。文本字幕处理时间与第一种方法的文本字幕处理时间相同。即, 考虑到在文本字幕处理单元 220 中对字幕执行解码所花费的时间 T_{decoding} 和将渲染的字幕从对象缓冲器 234 输出到图形面板器 240 所花费的时间 $T_{\text{composition}}$, 必须快速执行对输出字幕的文本字幕处理。然而, 在该方法中, 由于存在多个对象, 所以通过将渲染
30 各对象所花费的时间相加来获得渲染时间。即, 方程 2 计算渲染时间。

方程 2

$$\begin{aligned}
 T_{\text{start}} - T_{\text{arrival}} &\geq T_{\text{decoding}} + T_{\text{compositioin}} \\
 T_{\text{decoding}} &= T_{\text{rendering}} + T_{\text{composition information generation}} \\
 T_{\text{rendering}} &= \sum_{i=0}^{\text{对象的个数}} T_{\text{OBJ}(i)} \\
 T_{\text{OBJ}} &= \sum_{i=0}^{\text{字符的个数}} T_{\text{char}(i)}
 \end{aligned}$$

5 在第二种方法中可存储到对象缓冲器 234 中的字幕文本的字符的个数所受到的限制和第一种方法中一样。

图 8A 至 8C 是示出通过为多个字幕分配多个组成信息数据以生成图像以使将一个图像对象包括在一个组成信息数据中的过程的示图。

10 在图 8A 中，样式‘Script1’、‘Script2’和‘Script3’被定义为用于字幕文本渲染的样式信息。参照图 8A，所述三个样式的每一个都使用字体‘Arial.ttf’、文本颜色‘black’、背景颜色‘white’、字符大小‘16pt’、对齐方法‘center’、输出方向‘left-to-right-top-to-bottom’和行高‘40px’。作为字幕文本参考位置，‘Script1’具有坐标(x1, y1)，‘Script2’具有坐标(x2, y2)，‘Script3’具有坐标(x3, y3)。作为文本输出区域，‘Script1’具有‘left1, top1, width1, height1’，‘Script2’具有‘left2, top2, width2, height2’，‘Script3’具有‘left3, top3, width3, height3’。

15 在图 8B 中，定义了通过使用样式‘Script1’、‘Script2’和‘Script3’渲染的字幕文本 810、820 和 830。参照图 8B，字幕文本 Hello 810 使用样式‘Script1’，并从‘00:10:00’到‘00:15:00’被输出，字幕文本 Subtitle 820 使用样式‘Script2’，并从‘00:12:00’到‘00:17:00’被输出，字幕文本 World 830 使用样式‘Script3’，并从‘00:14:00’到‘00:19:00’被输出。因此，在‘00:12:00’和‘00:17:00’之间输出了两个或三个字幕文本。

20 图 8C 显示输出在图 8A 和 8B 中定义的字幕的结果。参照图 8C，将详细描述每个示出的时间窗口中的文本字幕处理单元 220 的每个缓冲器中存储的数据。

25 从‘00:10:00’：当输出的组成信息包括空字幕图像时，文本布局/字体渲染器 222 包括：

元素控制数据缓冲器：空；

文本数据缓冲器: 空;

样式数据缓冲器: 空; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

5 从‘00:10:00’: 当输出的组成信息包括在其中渲染了字幕文本 Hello 810 的图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 字幕文本 Hello 810 的加载控制信息;

文本数据缓冲器: ‘Hello’;

样式数据缓冲器: 样式信息‘Script1’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

10 从‘00:12:00’: 当输出的组成信息包括在其中渲染了字幕文本 Hello 810 和字幕文本 Subtitle 820 的图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 字幕文本 Subtitle 820 的加载控制信息;

文本数据缓冲器: ‘Subtitle’;

样式数据缓冲器: 样式信息‘Script2’; 和

15 字体数据缓冲器: 字体信息‘Arial.ttf’。

从‘00:14:00’: 当输出的组成信息包括在其中渲染了字幕文本 Hello 810、字幕文本 Subtitle 820 和字幕文本 World 830 的图像时, 文本布局/字体渲染器 222 包括:

元素控制数据缓冲器: 字幕文本 World 830 的加载控制信息;

20 文本数据缓冲器: ‘World’;

样式数据缓冲器: 样式信息‘Script3’; 和

字体数据缓冲器: 字体信息‘Arial.ttf’。

‘00:15:00’之后: 在准备在‘00:19:00’之后将被输出的后面的字幕文本的输出之前, 文本字幕处理单元 220 不执行任何操作。因此, 通过表现引擎 230 25 控制从文本字幕处理单元 220 接收的字幕‘Hello’、‘Subtitle’和‘World’的组成信息来执行‘00:15:00’和‘00:19:00’之间的字幕输出的改变。

即, 在‘00:15:00’, 表现引擎 230 从组成缓冲器 233 和对象缓冲器 234 删除字幕‘Hello’的组成信息和位图图像对象, 仅将字幕‘Subtitle’和‘World’的组成信息输出到屏幕上。在‘00:17:00’, 表现引擎 230 从组成缓冲器 233 和对象 30 缓冲器 234 删除字幕‘Subtitle’的组成信息和位图对象, 仅将字幕‘World’的组成信息输出到屏幕上。另外, 在‘00:19:00’, 表现引擎 230 从组成缓冲器 233

和对象缓冲器 234 删除字幕‘World’的组成信息和位图对象，并且不再将字幕输出到屏幕上。

在上述第三种方法中，通过将不同的样式应用到具有重叠输出时间的多个字幕文本来为每个字幕文本产生一个字幕图像，为每个字幕图像产生一个组成信息数据，并将产生的多个组成信息数据发送到表现引擎 230。文本字幕处理时间与第一种方法的文本字幕处理时间相同。由于在第一种和第二种方法中用于具有重叠输出时间的多个字幕文本的一个组成信息数据被组成并被输出，所以在第一种方法和第二种方法中仅考虑一个组成信息数据的处理时间，由于在第三种方法中每个字幕文本组成独立的组成信息数据，所以在第三种方法中产生并输出多个组成信息数据。因此，对于第三种方法的字幕文本处理开始时间，必须考虑最坏的情况，即，用于具有相同的输出开始时间的多个字幕的多个组成信息数据被同时产生并被输出的情况。这由方程 3 来描述。

方程 3

$$T_{\text{start}} - T_{\text{arrival}} \geq T_{\text{decoding}} + T_{\text{composition}}$$

$$T_{\text{decoding}} = T_{\text{rendering}} + T_{\text{composition information generation}}$$

$$T_{\text{composition information generation}} = \sum_{i=0}^{\text{组成信息数据的个数}} T_{\text{composition information}(i)}$$

$$T_{\text{rendering}} = \sum_{i=0}^{\text{对象的个数}} T_{\text{OBJ}(i)}$$

$$T_{\text{OBJ}} = \sum_{i=0}^{\text{字符的个数}} T_{\text{char}(i)}$$

通过将每一个字幕的组成信息产生时间 $T_{\text{composition information}}$ 加在一起来获得产生多个组成信息数据所花费的时间 $T_{\text{composition information generation}}$ 。通过将每一个字幕的渲染时间 T_{OBJ} 加在一起来获得通过渲染多个字幕产生多个对象所花费的时间 $T_{\text{rendering}}$ 。通过将每一个相关字幕中包括的每个字符的渲染时间 T_{char} 加在一起来获得渲染一个字幕所花费的时间 T_{OBJ} 。参照方程 3，为了同时输出多个包括多个字符的字幕，渲染在所述字幕中包括的所有字符所花费的时间、组成多个组成信息数据、并且输出所述多个组成信息所花费的时间的总和必须小于文本字幕处理单元 220 的字幕输出时间和字幕处理开始时间之间的差。

在第三种方法中可存储到对象缓冲器 234 中的字幕文本的字符的个数所受到的限制和第一种方法和第二种方法中一样。

如第三种方法所述，在信息存储介质和用支持同时输出多个组成信息数据的结构构成的再现设备中，可将文本字幕和另一位图图像同时输出到屏幕上。

在 AV 流中被压缩和编码的数据包括视频数据、音频数据、基于位图的字幕和其他非字幕的位图图像。为了指示是针对超过 14 岁的人的 TV 节目而显示在屏幕的右上方的图像‘TV-14’是非字幕位图图像的示例。在传统方法中，由于在一个时间点仅一个组成信息数据被显示在屏幕上，所以为了同时输出位图字幕和非字幕位图图像，需要在组成信息中分开定义用于输出位图字幕的区域和用于输出非字幕位图图像的区域。

因此，当用户由于不需要字幕的输出而关闭字幕的输出时，解码器仅停止对字幕解码。因此，由于不将字幕数据发送到对象缓冲器，所以字幕从屏幕上消失，仅非字幕位图图像被连续输出到屏幕上。

当文本字幕处理单元 220 通过使用一个组成信息数据为字幕产生图像并将该组成信息数据发送到表现引擎 230 以输出该字幕时，如果字幕的输出被关闭，则在 AV 流中记录的非字幕位图也不被输出。因此，在如本发明的第三种方法中所述的可将多个组成信息数据同时输出到屏幕上的情况下，当选择了文本字幕而非位图字幕时，可连续输出 AV 流中包括的组成信息数据中的除位图字幕之外的图像，并可通过使用由字幕处理单元 220 产生的组成信息来输出文本字幕。即，可将文本字幕和其他非字幕位图图像同时输出到屏幕上。

在通用计算机中可通过从计算机可读介质运行程序来实现本发明，所述计算机可读介质包括但不限于存储介质，例如磁性存储介质(ROM、RAM、软盘、磁盘等)、光学可读介质(CD-ROM、DVD 等)和载波(通过互联网的传输)。为了使经网络连接的几个计算机系统实现分布式处理，本发明可被实现为包括计算机可读程序代码单元的计算机可读介质。可由本发明所属领域的程序员容易地推出用于实现本发明的函数程序、代码和代码段。

尽管已经显示和描述了本发明的一些实施例，但是本领域的技术人员应该理解，在不脱离由权利要求及其等同物限定其范围的本发明的原则和精神的情况下，可对这些实施例进行改变。

图 1

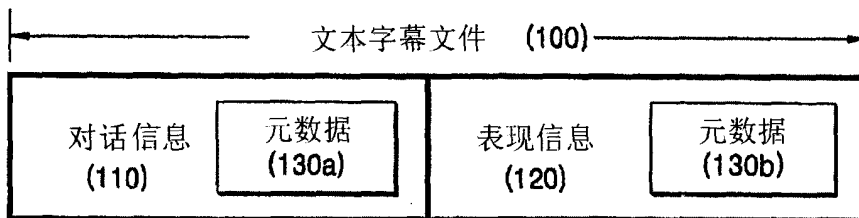


图 2

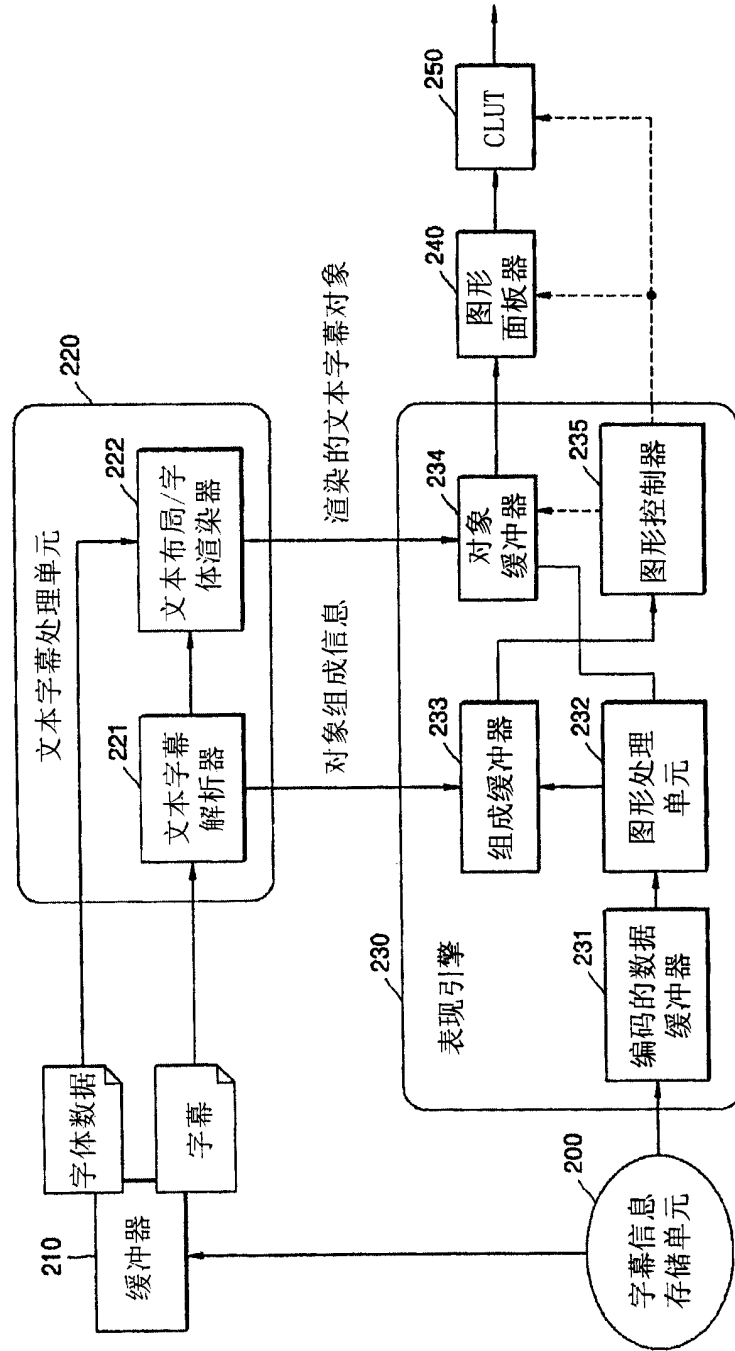


图 3

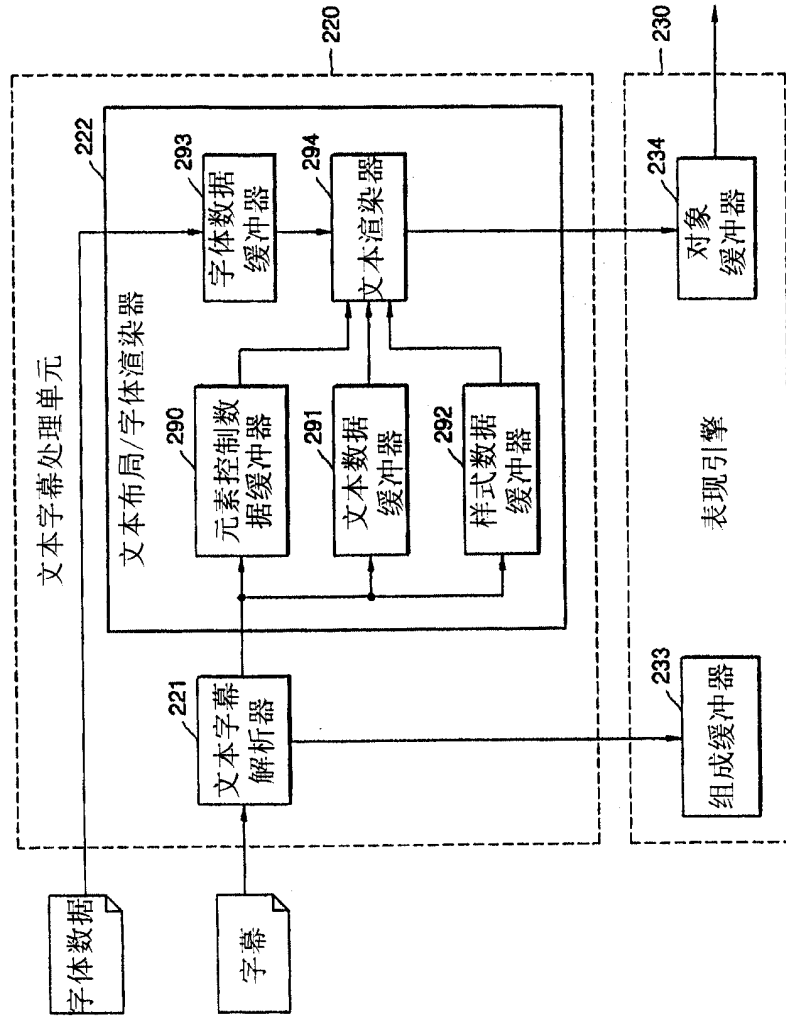


图 4

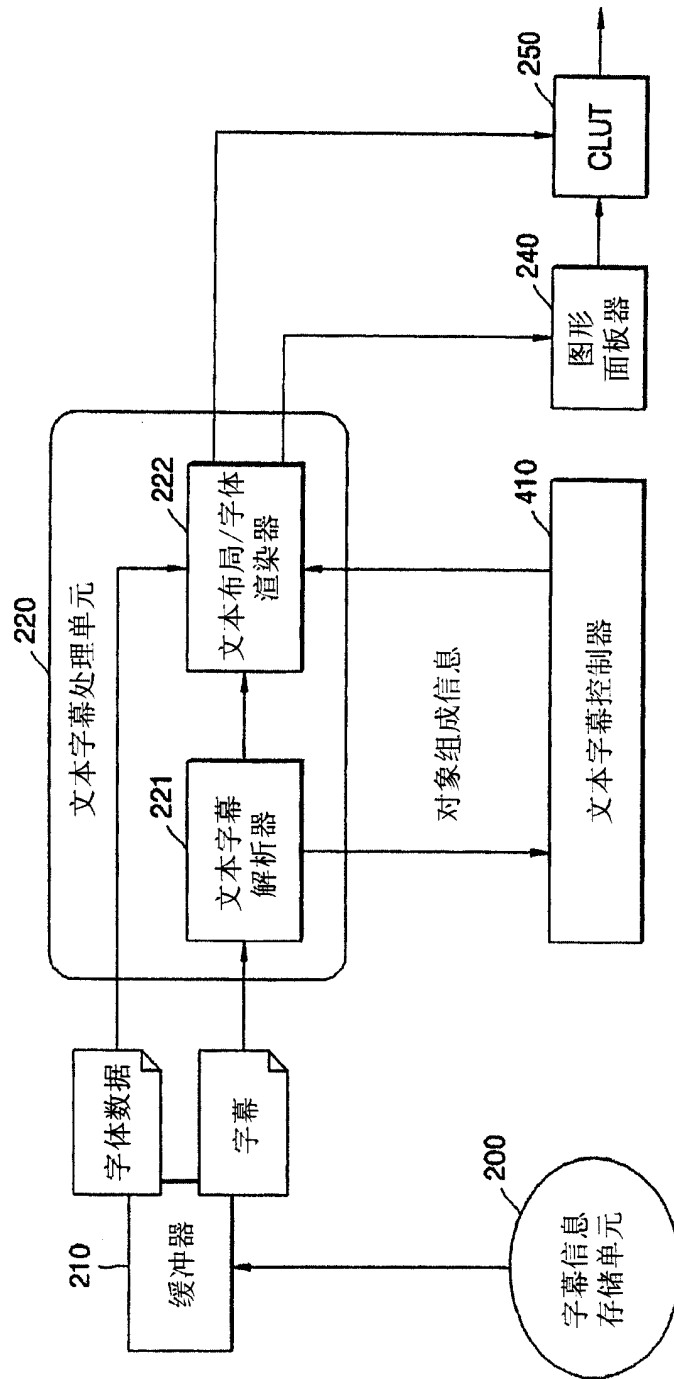


图 5

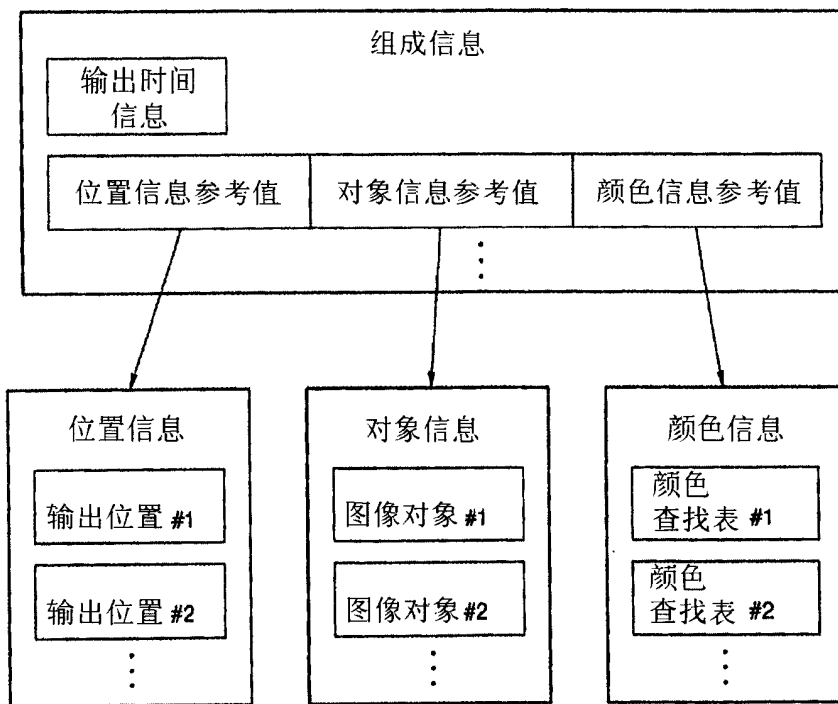


图 6A

```
<style name="Script" font="Arial.ttf" color="black"
background="white" size="16pt" position="x, y" align="center"
direction="left-to-right-top-to-bottom" region="left, top, width,
height" line-height="40px" />
```

图 6B

```
<text start="00:10:00" end="00:15:00" style="Script"> } 610
    Hello
</text>
<text start="00:12:00" end="00:17:00" style="Script"> } 620
    <br/>Subtitle
</text>
<text start="00:14:00" end="00:19:00" style="Script"> } 630
    <br/><br/>World
</text>
```

图 6C

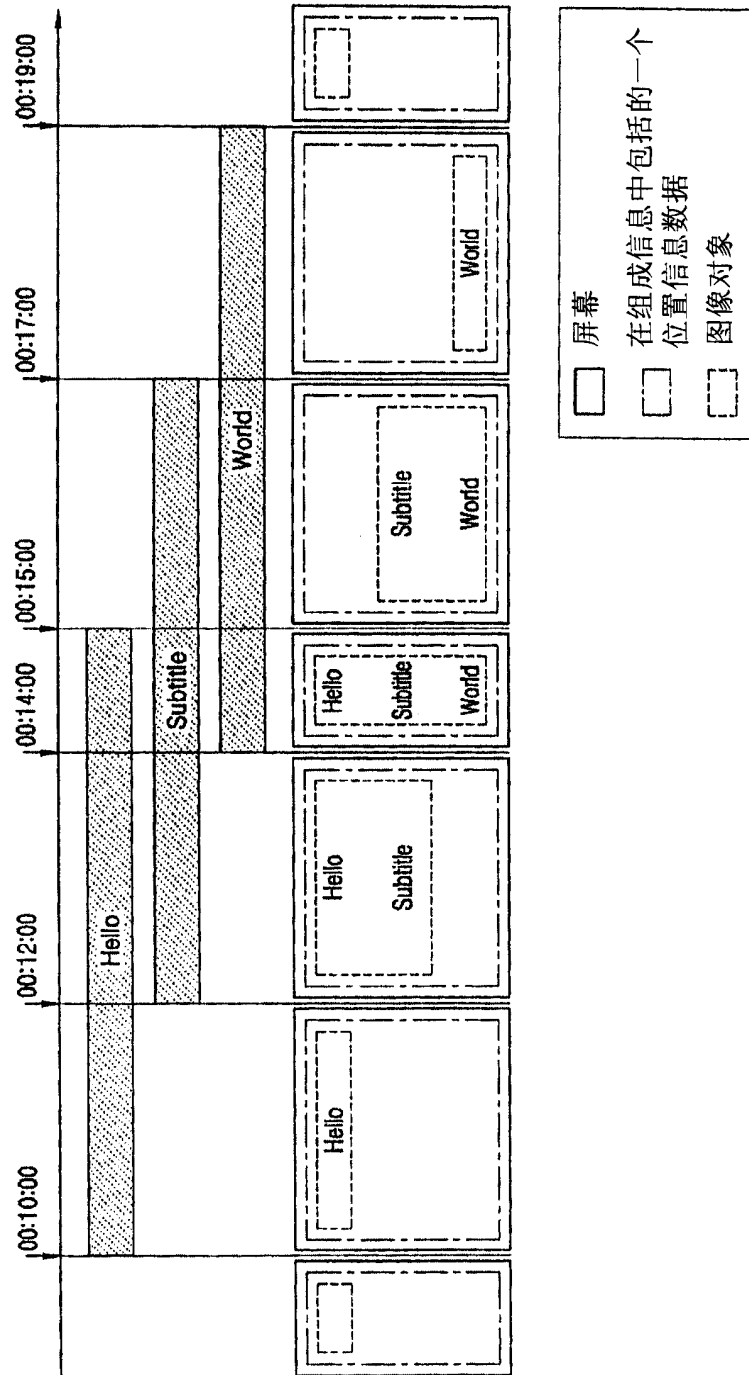


图 7A

```
<style name="Script1" font="Arial.ttf" color="black"
background="white" size="16pt" position="x1, y1" align="center"
direction="left-to-right-top-to-bottom" region="left1, top1, width1,
height1" line-height="40px" />
```

```
<style name="Script2" font="Arial.ttf" color="black"
background="white" size="16pt" position="x2, y2" align="center"
direction="left-to-right-top-to-bottom" region="left2, top2, width2,
height2" line-height="40px" />
```

```
<style name="Script3" font="Arial.ttf" color="black"
background="white" size="16pt" position="x3, y3" align="center"
direction="left-to-right-top-to-bottom" region="left3, top3, width3,
height3" line-height="40px" />
```

图 7B

```
<text start="00:10:00" end="00:15:00" style="Script1">
    Hello } 710
</text>
<text start="00:12:00" end="00:17:00" style="Script2">
    Subtitle } 720
</text>
<text start="00:14:00" end="00:19:00" style="Script3">
    World } 730
</text>
```

图 7C

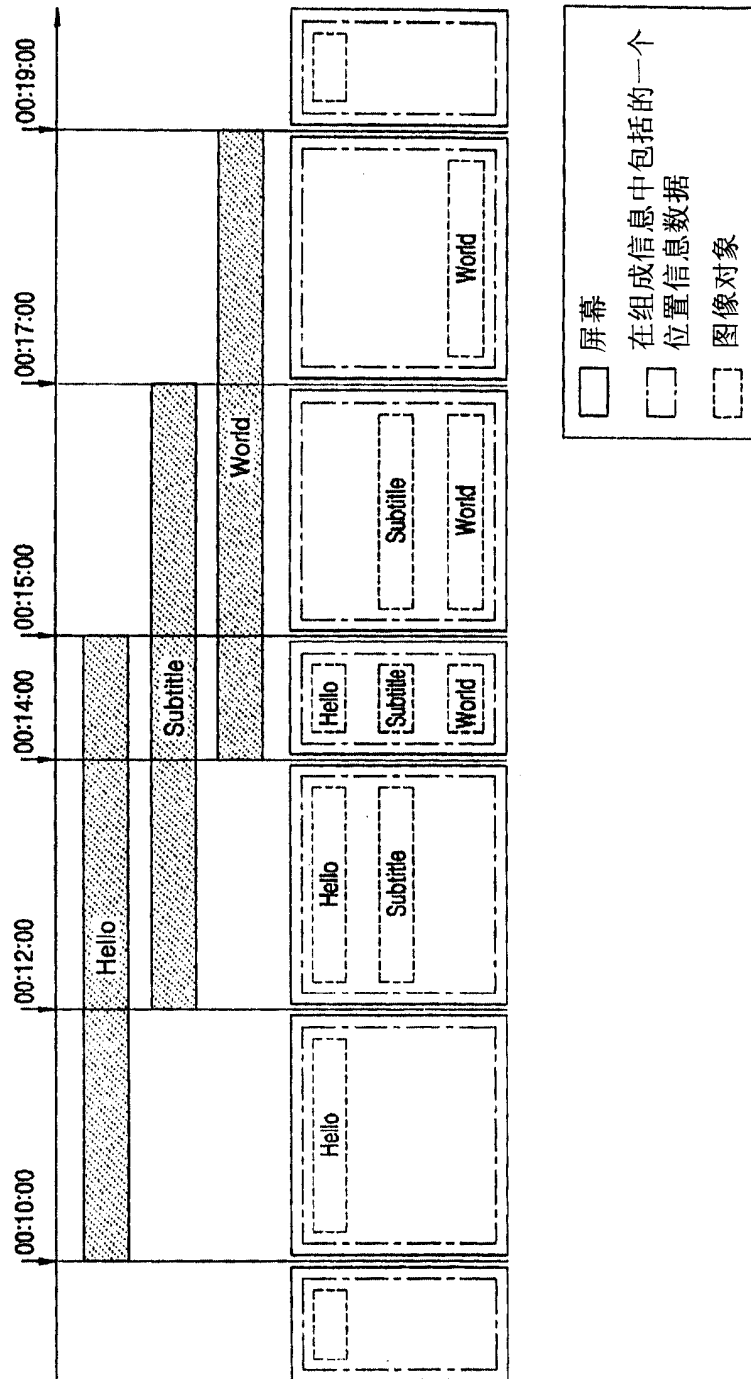


图 8A

```
<style name="Script1" font="Arial.ttf" color="black"
background="white" size="16pt" position="x1, y1" align="center"
direction="left-to-right-top-to-bottom" region="left1, top1, width1,
height1" line-height="40px" />
```

```
<style name="Script2" font="Arial.ttf" color="black"
background="white" size="16pt" position="x2, y2" align="center"
direction="left-to-right-top-to-bottom" region="left2, top2, width2,
height2" line-height="40px" />
```

```
<style name="Script3" font="Arial.ttf" color="black"
background="white" size="16pt" position="x3, y3" align="center"
direction="left-to-right-top-to-bottom" region="left3, top3, width3,
height3" line-height="40px" />
```

图 8B

```
<text start="00:10:00" end="00:15:00" style="Script1">
    Hello } 810
</text>
<text start="00:12:00" end="00:17:00" style="Script2">
    Subtitle } 820
</text>
<text start="00:14:00" end="00:19:00" style="Script3">
    World } 830
</text>
```

图 8C

