

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.  
G06F 15/00 (2006.01)



# [12] 发明专利申请公布说明书

[21] 申请号 200480041452.9

[43] 公开日 2008年2月13日

[11] 公开号 CN 101124561A

[22] 申请日 2004.12.8

[21] 申请号 200480041452.9

[30] 优先权

[32] 2003.12.8 [33] US [31] 10/731,809

[86] 国际申请 PCT/US2004/041667 2004.12.8

[87] 国际公布 WO2005/057906 英 2005.6.23

[85] 进入国家阶段日期 2006.8.7

[71] 申请人 DIVX 公司

地址 美国加利福尼亚州

[72] 发明人 A·范佐斯特 J·布拉利斯

C·德拉姆 T·赛瑞 B·科特

D·N·汤普森 J·罗塔

A·布尔克 A·梅奥

E·W·格拉伯 J·唐纳森

H·希契科克 D·查瓦锐额

M·佛洛伊德 A·巴特 B·福吉

A·鲁宾逊

[74] 专利代理机构 北京纪凯知识产权代理有限公司

代理人 赵蓉民 薛峰

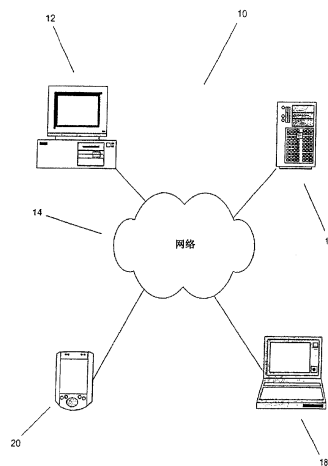
权利要求书 23 页 说明书 78 页 附图 29 页

[54] 发明名称

多媒体分发系统

[57] 摘要

本发明描述了一种多媒体文件，以及产生、分发和使用该多媒体文件的方法。根据本发明实施例的多媒体文件能够包含多个视频轨、多个音频轨、多个字幕轨、能够被用来产生菜单界面以访问文件内容的数据，以及关于所述文件的内容的“元数据”。根据本发明若干实施例的多媒体文件还包括对所述文件之外的视频轨、音频轨、字幕轨和“元数据”的引用。根据本发明的多媒体文件的一个实施例，包括多个编码视频轨。



1. 一种根据本发明的多媒体文件，其包括多个编码视频轨。
2. 根据权利要求 1 所述的多媒体文件，其中所述多媒体文件包括多个连接“RIFF”块。
3. 根据权利要求 2 所述的多媒体文件，其中每个编码视频轨被包含在一个分离的“RIFF”块中。
4. 根据权利要求 1 所述的多媒体文件，其中所述视频是利用心理视觉增强来编码的。
5. 根据权利要求 1 所述的多媒体文件，其中每个视频轨具有至少一个与之关联的音频轨。
6. 根据权利要求 5 所述的多媒体文件，其中：  
每个视频轨被编码为“RIFF”块内的一系列“视频”块；以及  
伴随每个视频轨的音频轨被编码为一系列“音频”块，所述“音频”块被交错在包含关联视频轨的“视频”块的“RIFF”块之内。
7. 根据权利要求 6 所述的多媒体文件，其中：  
每个“视频”块包含能够被用来从视频轨中产生单一视频帧的信息；而且  
每个“音频”块包含来自于音频轨中与使用“视频”块所产生的所述帧相伴部分的音频信息。
8. 根据权利要求 7 所述的多媒体文件，其中所述“音频”块被交错在所述“RIFF”块内的相应“视频”块之前。
9. 一种用于对多媒体文件进行编码的系统，包括：  
处理器，其被配置成：

对多个视频轨进行编码；  
连接所述编码视频轨；以及  
将所述连接编码视频轨写入单一文件。

10. 根据权利要求 9 所述的系统, 其中所述处理器被配置成编码所述视频轨, 以致每个视频轨被包含在分离的“RIFF”块之内。

11. 根据权利要求 9 所述的系统, 其中所述处理器被配置成利用心理视觉增强来编码所述视频。

12. 根据权利要求 9 所述的系统, 其中每个视频轨具有至少一个与之关联的音频轨。

13. 根据权利要求 12 所述的多媒体文件, 其中:  
所述处理器被配置成:

将每个视频轨编码为“RIFF”块内的一系列“视频”块; 以及

将至少一个伴随着每个视频轨的音频轨编码为一系列“音频”块, 其在包含所述关联视频轨的“视频”块的“RIFF”块内交错。

14. 根据权利要求 13 所述的多媒体文件, 其中:  
所述处理器被配置成:

编码所述视频轨, 以致每个“视频”块包含能够用来从视频轨产生单一视频帧的信息; 以及

编码与视频轨相关联的所述音频轨, 以致每个“音频”块包含音频信息, 该音频信息来自于所述音频轨中的一部分, 该部分伴随着使用“视频”块所产生的帧, 该“视频”块产生自所述视频轨。

15. 根据权利要求 14 所述的多媒体文件, 其中所述处理器被配置成使每个“音频”块被交错在所述“RIFF”块内的相应“视频”块之前。

16. 一种用于解码包含多个编码视频轨的多媒体文件的系统，包括：

处理器，其被配置成从所述多媒体文件中提取信息；

其中所述处理器被配置成提取关于所述多媒体文件内所包含的编码视频轨数目的信息。

17. 根据权利要求 16 所述的系统，其中所述处理器被配置成在“RIFF”块内定位一个编码视频轨。

18. 根据权利要求 17 所述的系统，其中：

第一编码视频轨被包含在具有标准 4cc 码的第一“RIFF”块中；

第二视频轨被包含在具有专用 4cc 码的第二“RIFF”块中；而且

所述专用 4cc 码以一个标准 4cc 码的前两个字符作为它的最后两个字符。

19. 根据权利要求 17 所述的系统，其中每个编码视频轨被包含在一个分离的“RIFF”块中。

20. 根据权利要求 16 所述的系统，其中所述解码视频轨与所述多媒体文件创建中被编码的原始视频轨类似。

21. 根据权利要求 20 所述的系统，其中所述解码视频轨和所述原始视频轨之间差异中的至少一些位于所述视频轨各帧的暗部分中。

22. 根据权利要求 20 所述的系统，其中所述解码视频轨和所述原始视频轨之间差异中的至少一些位于所述视频轨的高速运动场景中。

23. 根据权利要求 16 所述的系统，其中每个视频轨具有至少一个与之相关联的音频轨。

24. 根据权利要求 23 所述的系统，其中：

所述处理器被配置成：

通过解码“RIFF”块内的一系列“视频”块从视频轨中显示视频；以及

通过解码交错在所述“RIFF”块内的一系列“音频”块，从伴随所述视频轨的音频轨中产生音频，其中所述“RIFF”块包含所述关联视频轨的所述“视频”块。

25. 根据权利要求 24 所述的多媒体文件，其中：

所述处理器被配置成：

使用从每个“视频”块中提取的信息，以产生所述视频轨的单一帧；以及

使用从每个“音频”块中提取的信息，以产生音频轨中与使用“视频”块所产生的所述帧相伴随的部分。

26. 根据权利要求 25 所述的多媒体文件，其中所述处理器被配置成将所述“音频”块定位在所述“RIFF”块中与所述“音频”块关联的“视频”块之前。

27. 一种根据本发明的多媒体文件，包括：

一系列的编码视频帧；

交错在所述编码视频帧之间的编码音频；

其中所述编码音频包括两个或更多个音频信息轨。

28. 根据权利要求 27 所述的多媒体文件，其中所述音频信息轨中的至少一个包括多个音频通道。

29. 根据权利要求 27 所述的多媒体文件，进一步包括头信息，其标识了所述多媒体文件中所包含的音频轨的数目。

30. 根据权利要求 27 所述的多媒体文件，进一步包括头信息，其包括关于所述音频信息轨中至少一个的描述信息。

31. 根据权利要求 27 所述的多媒体文件，其中：

每个编码视频帧之前有编码音频信息；以及  
所述视频帧之前的所述编码音频信息包括用于每个音频轨中伴随着所述编码视频帧的那部分的音频信息。

32. 根据权利要求 27 所述的多媒体文件，其中所述视频信息被存储为所述多媒体文件内的块。

33. 根据权利要求 32 所述的多媒体文件，其中每个视频信息块包括单一视频帧。

34. 根据权利要求 33 所述的多媒体文件，其中：  
所述音频信息被存储为所述多媒体文件内的块；  
来自两个分离的音频轨的音频信息不被包含在单一的音频信息块中。

35. 根据权利要求 34 所述的多媒体文件，其中所述视频块来自每个所述音频轨的至少一个音频块分离开。

36. 根据权利要求 35 所述的多媒体文件，其中分离所述视频块的所述音频块包含所述音频轨中伴随着所述视频信息的那些部分的音频信息，其中所述视频信息包含在所述音频块之后的视频块内。

37. 一种用于对多媒体文件进行编码的系统，包括：  
处理器，其被配置成：  
    对一个视频轨进行编码；  
    对多个音频轨进行编码；  
    对来自所述视频轨的信息与来自所述多个音频轨的信息进行交错；并且  
    将交错后的视频和音频信息写入单一文件。

38. 根据权利要求 37 所述的系统，其中至少一个所述音频轨包括多个音频通道。

39. 根据权利要求 37 所述的系统，其中所述处理器被配置成对标识了编码音频轨数目的头信息进行编码，并将所述头信息写入所述单一文件。

40. 根据权利要求 37 所述的系统，其中所述处理器被进一步配置成对头信息进行编码，并将所述头信息写入所述单一文件，其中所述头信息标识了关于至少一个所述编码音频轨的描述信息。

41. 根据权利要求 37 所述的系统，其中所述处理器将所述视频轨编码为视频信息块。

42. 根据权利要求 41 所述的系统，其中所述处理器将每个音频轨编码为一系列的音频信息块。

43. 根据权利要求 42 所述的系统，其中每个音频信息块包含来自单一音频轨的音频信息。

44. 根据权利要求 43 所述的系统，其中所述处理器被配置成在视频信息块之间交错音频信息块。

45. 根据权利要求 44 所述的系统，其中：

所述处理器被配置成，将每个音频轨中伴随着一个“视频”块中视频信息的那个部分编码在“音频”块中；而且

所述处理器被配置成将所述“视频”块与所述“音频”块交错，以致每个“视频”块之前为“音频”块，这些“音频”块所包含的来自每个所述音频轨的音频信息伴随着所述“视频”块中所包含的视频信息。

46. 根据权利要求 45 所述的系统，其中所述处理器被配置成编码所述视频轨，以致单一视频帧被包含在每个视频块之内。

47. 根据权利要求 45 所述的系统，其中所述处理器是通用处理器。

48. 根据权利要求 45 所述的系统，其中所述处理器是专用电路。

49. 一种解码包含多个音频轨的多媒体文件的系统，包括：

处理器，其被配置成从所述多媒体文件中提取信息；

其中所述处理器被配置成提取信息，该信息关于包含在所述多媒体文件内的音频轨数目。

50. 根据权利要求 49 所述的系统，其中：

所述处理器被配置成从所述多个音频轨选择单一音频轨；

所述处理器被配置成解码来自所选音频轨的音频信息。

51. 根据权利要求 49 所述的系统，其中所述音频轨中的至少一个包含了多个音频通道。

52. 根据权利要求 49 所述的系统，其中所述处理器被配置成从所述多媒体文件内的头中提取信息，该多媒体文件内的头包含了有关至少一个所述音频轨的描述信息。

53. 一种用于传递多媒体信息的系统，包括：

网络；

存储装置，其包含了多媒体文件并通过服务器连接到所述网络；

连接到所述网络的客户机；

其中所述客户机能够请求从所述服务器传送所述多媒体文件；而

且

其中所述多媒体文件包括至少一个视频轨和多个伴随所述视频轨的音频轨。

54. 一种根据本发明的多媒体文件，包括：

一系列编码视频帧；以及

至少一个编码字幕轨，其在所述编码视频帧之间交错。



55. 根据权利要求 54 所述的多媒体文件，其中所述至少一个编码字幕轨包括多个编码字幕轨。

56. 根据权利要求 54 所述的多媒体文件，进一步包括头信息，其标识了包含在所述多媒体文件中的编码字幕轨的数目。

57. 根据权利要求 54 所述的多媒体文件，进一步包括头信息，其包含了有关编码字幕轨的至少一个的描述信息。

58. 根据权利要求 54 所述的多媒体文件，其中每个字幕轨包括一系列位图。

59. 根据权利要求 58 所述的多媒体文件，其中每个字幕轨包括一系列压缩位图。

60. 根据权利要求 59 所述的多媒体文件，其中每个位图是使用游程长度编码来压缩的。

61. 根据权利要求 54 所述的多媒体文件，其中：  
所述一系列编码视频帧被编码为一系列视频块；  
每个编码字幕轨被编码为一系列字幕块；  
其中每个字幕块包括了信息，其能够作为文本显示在显示器上的。

62. 根据权利要求 61 所述的多媒体文件，其中每个字幕块包含关于单一字幕的信息。

63. 根据权利要求 62 所述的多媒体文件，其中每个字幕块包括关于视频序列中字幕应被叠加在其上的那个部分的信息。

64. 根据权利要求 62 所述的多媒体文件，其中每个字幕块包括关于显示器中字幕应被定位在其中的那个部分的信息。

65. 根据权利要求 62 所述的多媒体文件，其中每个字幕块包括关于所述字幕颜色的信息。

66. 根据权利要求 65 所述的多媒体文件，其中所述关于所述颜色的信息包括调色板。

67. 根据权利要求 66 所述的多媒体文件，其中所述字幕块包含第一字幕块和第二字幕块，所述第一字幕块包括关于第一调色板的信息，所述第二字幕块包括关于第二调色板的信息，其接替关于所述第一调色板的信息。

68. 一种用于编码多媒体文件的系统，包括：

处理器，其被配置成：

编码视频轨；

编码至少一个字幕轨；

将来自所述视频轨的信息与来自所述至少一个字幕轨的信息相交错；以及

将交错的视频和字幕信息写入单一文件中。

69. 根据权利要求 68 所述的系统，其中所述至少一个字幕轨包含多个字幕轨。

70. 根据权利要求 68 所述的系统，其中所述处理器被进一步配置成对头信息进行编码并将其写入所述单一文件中，所述头信息标识了包含在所述多媒体文件中的字幕轨的数目。

71. 根据权利要求 68 所述的系统，其中所述处理器被进一步配置成对关于至少一个所述字幕轨的描述信息进行编码，并将其写入所述单一文件中。

72. 根据权利要求 68 所述的系统，其中：

所述视频轨被编码为视频块；以及  
所述至少一个字幕轨中的每一个都被编码为字幕块。

73. 根据权利要求 72 所述的系统，其中：  
每个所述字幕块包含单一字幕，其伴随着所述视频轨的一部分；  
所述交错器被配置成将每个字幕块交错在所述视频块之前，所述视频块包含视频轨中的、所述字幕轨之内的字幕与之相伴随的部分。

74. 根据权利要求 72 所述的系统，其中所述处理器被配置成通过将字幕编码为位图来产生字幕块。

75. 根据权利要求 72 所述的系统，其中所述字幕被编码为压缩位图。

76. 根据权利要求 75 所述的系统，其中所述位图是使用游程长度编码来压缩的。

77. 根据权利要求 76 所述的系统，其中所述处理器在每个字幕块中包括了关于视频序列中所述字幕应被叠加在其上的那部分的信息。

78. 根据权利要求 73 所述的系统，其中所述处理器在每个字幕块中包含了关于显示器中所述字幕应被定位在其上的那部分的信息。

79. 根据权利要求 73 所述的系统，其中所述处理器在每个字幕块中包括关于所述字幕颜色的信息。

80. 根据权利要求 73 所述的系统，其中所述关于所述颜色的信息包括调色板。

81. 根据权利要求 80 所述的系统，其中所述字幕块包含第一字幕块和第二字幕块，所述第一字幕块包括了关于第一调色板的信息，所述第二字幕块包括了关于第二调色板的信息，其接替了关于所述第一

调色板的信息。

82. 一种用于解码多媒体文件的系统，其包括：  
处理器，其被配置成从所述多媒体文件中提取信息；  
其中所述处理器被配置成检查所述多媒体文件以确定是否有至少一个字幕轨。

83. 根据权利要求 82 所述的系统，其中所述至少一个字幕轨包括多个字幕轨，而且所述处理器被配置成确定所述多媒体文件中字幕轨的数目。

84. 根据权利要求 82 所述的系统，其中所述处理器被进一步配置成提取头信息，其标识了来自所述多媒体文件字幕轨的数目。

85. 根据权利要求 82 所述的系统，其中所述处理器被进一步配置成提取关于来自所述多媒体文件的至少一个所述字幕轨的描述信息。

86. 根据权利要求 82 所述的用于解码多媒体文件的系统，其中：  
所述多媒体文件包括至少一个视频轨，其被编码为视频块；以及  
所述多媒体文件包括至少一个字幕轨，其被编码为字幕块。

87. 根据权利要求 86 所述的用于解码多媒体文件的系统，其中每个字幕块包含关于单一字幕的信息。

88. 根据权利要求 87 所述的用于解码多媒体文件的系统，其中：  
每个字幕在字幕块中被编码为位图；  
所述处理器被配置成解码所述视频轨；以及  
所述处理器被配置成通过在所述视频序列的一部分上叠加所述位图，来构建用于显示的视频帧。

89. 根据权利要求 88 所述的用于解码多媒体文件的系统，其中：  
所述字幕被编码为压缩位图；以及

所述处理器被配置成解压缩所述位图。

90. 根据权利要求 89 所述的用于解码多媒体文件的系统，其中所述处理器被配置成解压缩游程长度编码的位图。

91. 根据权利要求 88 所述的用于解码多媒体文件的系统，其中：  
每个视频块包括关于所述视频轨中所述字幕应被叠加在其上的那部分的信息；而且

所述处理器被配置成通过在每一视频帧上叠加所述字幕的所述位图，来产生视频帧序列，所述在每一视频帧上叠加所述字幕的所述位图由所述字幕块中的信息来指示。

92. 根据权利要求 88 所述的用于解码多媒体文件的系统，其中：  
每个字幕块包括关于所述字幕应被定位在帧内的位置的信息；以及

所述处理器被配置成在所述字幕块内的所述信息所指示的每个视频帧内的位置上叠加所述字幕。

93. 根据权利要求 88 所述的用于解码多媒体文件的系统，其中：  
每个字幕块包括了关于所述字幕颜色的信息；以及  
所述处理器被配置成采用所述字幕块内的所述颜色信息指示的一种颜色或多种颜色来叠加字幕。

94. 根据权利要求 93 所述的用于解码多媒体文件的系统，其中：  
所述字幕块内的所述颜色信息包括调色板；以及  
所述处理器被配置成使用所述调色板来获得使用在所述字幕的所述位图中的颜色信息来叠加所述字幕。

95. 根据权利要求 94 所述的系统，其中所述字幕块包括第一字幕块和第二字幕块，所述第一字幕块包含关于第一调色板的信息，所述第二字幕块包括关于第二调色板的信息；而且  
其中所述处理器：

被配置成在所述第一块被处理之后叠加所述字幕，其使用了所述第一调色板以获得信息，该信息关于用在所述字幕的所述位图中的颜色。

被配置成在所述第二块被处理之后叠加所述字幕，其使用了所述第二调色板以获得信息，该信息关于用在所述字幕的所述位图中的颜色。

96. 一种用于传递多媒体信息的系统，包括：

网络；

存储装置，其包含了多媒体文件并通过服务器连接到所述网络；

连接到所述网络的客户机；

其中所述客户机能够从所述服务器请求传送所述多媒体文件；以及

其中所述多媒体文件包括至少一个视频轨和至少一个伴随所述视频轨的字幕轨。

97. 一种根据本发明的多媒体文件，其包括：

一系列编码视频帧；以及

编码菜单信息。

98. 根据权利要求 97 所述的多媒体文件，其中所述编码菜单信息被存储为块。

99. 根据权利要求 98 所述的多媒体文件，进一步包括菜单信息的至少两个分离的“Menu”块。

100. 根据权利要求 99 所述的多媒体文件，其中所述至少两个分离的“Menu”块被包含在至少两个分离的“RIFF”块中。

101. 根据权利要求 100 所述的多媒体文件，其中包含“Menu”块的第一“RIFF”块包括标准 4cc 码，以及包含“Menu”块的第二“RIFF”块包括专用 4cc 码，在其中标准 4cc 码的前两个字符表现为专用 4cc 码

的最后两个字符。

102. 根据权利要求 99 所述的多媒体文件，其中至少两个分离的“Menu”块被包含在一个单一的“RIFF”块中。

103. 根据权利要求 99 所述的多媒体文件，其中所述“Menu”块包括了描述一系列菜单的块和一个“MRIF”块，该“MRIF”块包含了与所述一系列菜单关联的媒体。

104. 根据权利要求 103 所述的多媒体文件，其中所述“MRIF”块包含媒体信息，其包括视频轨、音频轨和覆盖轨。

105. 根据权利要求 103 所述的多媒体文件，其中：  
所述描述一系列菜单的块包括：

描述总菜单系统的块；

根据语言分组菜单的至少一个块；

描述独立的菜单显示及其伴随的背景音频的至少一个块；

描述了菜单上的按钮的至少一个块；

描述了所述按钮在屏幕上的位置的至少一个块；以及

描述了与一个按钮关联的各种动作的至少一个块。

106. 根据权利要求 97 所述的多媒体文件，进一步包括：  
到第二文件的链接；  
其中编码菜单信息被包含在所述第二文件之内。

107. 一种用于编码多媒体文件的系统，包括：  
处理器，其被配置成编码菜单信息；  
其中所述处理器还被配置成产生包含编码视频轨和所述编码菜单信息的多媒体文件。

108. 根据权利要求 107 所述的系统，其中：  
所述处理器被配置成：

产生所述菜单的对象模型；  
将所述对象模型转换为配置文件；  
语法分析所述配置文件为块；  
产生包含媒体信息的 AVI 文件；  
将 AVI 文件中的媒体交错到“MRIF”块中；以及  
将语法分析后的块和所述“MRIF”块连接，以创建一个“Menu”  
块。

109. 根据权利要求 108 所述的系统，其中所述处理器进一步被配置成使用所述对象模型来产生一第二较小“Menu”块。

110. 根据权利要求 107 所述的系统，其中所述处理器被配置成编码第二菜单。

111. 根据权利要求 110 所述的系统，其中所述处理器将所述第一编码菜单插入到第一“RIFF”块，以及将所述第二编码菜单插入到第二“RIFF”块。

112. 根据权利要求 110 所述的系统，其中所述处理器使得所述第一和第二编码菜单包含在一个单一的“RIFF”块中。

113. 根据权利要求 107 所述的系统，其中所述处理器被配置成在所述多媒体文件中插入第二文件中的编码菜单的引用。

114. 一种用于解码多媒体文件的系统，包括：  
处理器，其被配置成从所述多媒体文件中提取信息；  
其中所述处理器被配置成检查所述多媒体文件，从而确定该多媒体文件中是否包含了编码菜单信息。

115. 根据权利要求 114 所述的系统，其中所述处理器被配置成从“RIFF”块内的“Menu”块中提取菜单信息。



116. 根据权利要求 115 所述的系统，其中所述处理器被配置成使用存储在所述“Menu”块中的视频信息，来构建菜单显示。

117. 根据权利要求 115 所述的系统，其中所述处理器被配置成使用存储在“Menu”块中的音频信息来产生伴随菜单显示的背景音频。

118. 根据权利要求 115 所述的系统，其中所述处理器被配置成通过将来自“Menu”块的覆盖物覆盖到来自“Menu”块的视频信息之上，来产生菜单显示。

119. 一种用于传输多媒体信息的系统，包括：  
网络；  
存储装置，其包含了多媒体文件并通过服务器连接到所述网络；  
连接到所述网络的客户机；  
其中所述客户机能够请求来自服务器的所述多媒体文件传送；以及  
其中所述多媒体文件包括编码菜单信息。

120. 一种多媒体文件，包括：  
一系列编码视频帧；  
有关所述多媒体文件的编码元数据；  
其中所述编码的元数据包括至少一个语句，其包括主题、谓词、对象和权限。

121. 根据权利要求 120 所述的多媒体文件，其中所述主题包含标识由所述元数据来描述的文件、项目、人员或组织的信息。

122. 根据权利要求 121 所述的多媒体文件，其中所述谓词包括指示主题特征的信息。

123. 根据权利要求 122 所述的多媒体文件，其中所述对象包括描述所述谓词所标识的主题特征的信息。

124. 根据权利要求 123 所述的多媒体文件，其中所述权限包括关于所述语句的源的信息。

125. 根据权利要求 120 所述的多媒体文件，其中：  
所述主题是包括了类型和数值的块；  
所述数值包含信息；以及  
所述类型指示所述块是资源还是匿名节点。

126. 根据权利要求 120 所述的多媒体文件，其中：  
所述谓词是一包括类型和数值的块；  
所述数值包含信息；以及  
所述类型指示所述数值信息是谓词 URI 还是顺序列表条目。

127. 根据权利要求 120 所述的多媒体文件，其中：  
所述对象是一包括类型、语言、数据类型和数值的块；  
所述数值包含信息；  
所述类型指示所述数值信息是 UTF-8 文字，还是整数文字或是文字 XML 数据；  
所述数据类型指示所述数值信息的类型；以及  
所述语言包含标识具体语言的信息。

128. 根据权利要求 127 所述的多媒体文件，其中：  
所述权限是一包括类型和数值的块；  
所述数值包含信息；以及  
所述类型指示所述数值信息是所述语句的所述权限。

129. 根据权利要求 120 所述的多媒体文件，其中所述被编码的数据中的至少一部分是以二进制数据来表现的。

130. 根据权利要求 120 所述的多媒体文件，其中所述被编码的数据中的至少一部分是以 64 位 ASCII 数据来表现。

131. 根据权利要求 120 所述的多媒体文件，其中所述被编码的数据的至少第一部分表现为二进制数据，而所述被编码的数据的至少第二部分表现为附加块，所述附加块包含表现为第二种格式的数据。

132. 根据权利要求 131 所述的多媒体文件，其中每个所述附加块包含单段元数据。

133. 一种用于编码多媒体文件的系统，其包括：

一处理器，其被配置成编码视频轨；

其中所述处理器还被配置成编码关于所述多媒体文件的元数据；

以及

其中所述编码的元数据包括至少一个语句，其包含主题、谓词、对象和权限。

134. 根据权利要求 133 所述的系统，其中所述主题包含标识由所述元数据来描述的文件、项目、人员或组织的信息。

135. 根据权利要求 134 所述的系统，其中所述谓词包括指示主题特征的信息。

136. 根据权利要求 135 所述的系统，其中所述对象包括描述所述谓词所标识的主题特征的信息。

137. 根据权利要求 136 所述的系统，其中所述权限包括关于所述语句的源的信息。

138. 根据权利要求 133 所述的系统，其中：

所述处理器被配置成将所述主题编码为块，该块包括类型和数值；

所述数值包含信息；以及

所述类型指示所述块是资源还是匿名节点。

139. 根据权利要求 133 所述的系统，其中：  
所述处理器被配置成将所述谓词编码为块，该块包括类型和数值；  
所述数值包含信息；以及  
所述类型指示所述数值信息是谓词 URI 还是顺序列表条目。

140. 根据权利要求 133 所述的系统，其中：  
所述处理器被配置成将所述对象编码为块，该块包括类型、语言、数据类型和数值；  
所述数值包含信息；  
所述类型指示所述数值信息是 UTF-8 文字，还是整数文字或是文字 XML 数据；  
所述数据类型指示所述数值信息的所述类型；以及  
所述语言包含标识具体语言的信息。

141. 根据权利要求 133 所述的系统，其中：  
所述处理器被配置成将所述权限编码为块，该块包括类型和数值；  
所述数值包含信息；以及  
所述类型指示所述数值信息是所述语句的所述权限。

142. 根据权利要求 133 所述的系统，其中所述处理器被进一步配置成将关于所述多媒体文件的元数据的至少一部分编码为二进制数据。

143. 根据权利要求 133 所述的系统，其中所述处理器被进一步配置成将关于所述多媒体文件的元数据的至少一部分编码为 64 位 ASCII 数据。

144. 根据权利要求 133 所述的系统，其中所述处理器被进一步配置成：  
编码关于所述多媒体文件的元数据的至少第一部分为二进制数据  
并

编码关于所述多媒体文件的元数据的至少第二部分为包含表现为第二种格式的数据的附加块。

145. 根据权利要求 144 所述的系统，所述处理器被进一步配置成用单段元数据编码所述附加块。

146. 一种用于解码多媒体文件的系统，其包括：

一处理器，其被配置成从所述多媒体文件中提取信息；

其中，所述处理器被配置成提取关于所述多媒体文件的元数据信息；以及

其中所述元数据信息包括至少一个语句，其包含主题、谓词、对象和权限。

147. 根据权利要求 146 所述的系统，其中所述处理器被配置成从所述主题中提取标识由所述元数据描述的文件、项目、人员或组织的信息。

148. 根据权利要求 147 所述的系统，其中所述处理器被配置成从所述谓词提取指示主题特征的信息。

149. 根据权利要求 148 所述的系统，其中所述处理器被配置成从所述对象中提取描述所述谓词所标识的主题特征的信息。

150. 根据权利要求 149 所述的系统，其中所述处理器被配置成从所述权限提取关于所述语句的源的信息。

151. 根据权利要求 146 所述的系统，其中：

所述主题是包括了类型和数值的块；并且

所述处理器被配置成通过检查所述类型来识别所述块包含了主题信息；以及

所述处理器被配置成从所述数值中提取信息。

152. 根据权利要求 146 所述的系统，其中：  
所述谓词是一包括类型和数值的块；并且  
所述处理器被配置成通过检查所述类型来识别所述块包含了谓词信息；以及  
所述处理器被配置成从所述数值中提取信息。

153. 根据权利要求 146 所述的系统，其中：  
所述对象是包括类型、语言、数据类型和数值的块；  
所述处理器被配置成通过检查所述类型来识别所述块是否包含对象信息；  
所述处理器被配置成检查所述数据类型以确定包含在所述数值中的信息的数据类型；  
所述处理器被配置成从所述数值中提取出所述数据类型所指示的类型的信息；以及  
所述处理器被配置成从所述语言中提取出标识具体语言的信息。

154. 根据权利要求 146 所述的方法，其中：  
所述权限是一包括类型和数值的块；并且  
所述处理器被配置成通过检查所述类型来识别所述块是否包含权限信息；以及  
所述处理器被配置成从所述数值中提取信息。

155. 根据权利要求 146 所述的系统，其中所述处理器被配置成从元数据语句中提取信息，并显示所述信息的至少一部分。

156. 根据权利要求 146 所述的系统，其中所述处理器被配置成使用所述元数据来构建数据结构，其指示了内存中定向标签图形。

157. 根据权利要求 146 所述的系统，其中所述处理器被配置成通过检查多个语句的所述主题、谓词、对象和权限中的至少之一，来从头到尾地搜索元数据以获得信息。

158. 根据权利要求 157 所述的系统，其中所述处理器被配置成作为图形用户界面的一部分来显示搜索的所述结果。

159. 根据权利要求 157 所述的系统，其中所述处理器被配置成响应外部设备的请求，执行搜索。

160. 根据权利要求 146 所述的系统，其中关于所述多媒体文件的元数据信息的至少一部分表现为二进制数据。

161. 根据权利要求 146 所述的系统，其中关于所述多媒体文件的元数据信息的至少一部分表现为 64 位 ASCII 数据。

162. 根据权利要求 146 所述的系统，其中关于所述多媒体文件的元数据信息的至少第一部分表现为二进制数据；且其中关于所述多媒体文件的元数据信息的至少第二部分表现为附加块，该块包含了表现为第二种格式的数据。

163. 根据权利要求 162 所述的系统，其中所述附加块包含单段元数据。

164. 一种用于传输多媒体信息的系统，包括：  
网络；  
存储装置，其包含了多媒体文件并通过服务器连接到所述网络；  
连接到所述网络的客户机；  
其中所述客户机能够请求从所述服务器传送所述多媒体文件；和  
其中所述多媒体文件包括关于所述多媒体文件的元数据；以及  
其中所述元数据包括至少一个语句，其包括主题、谓词、对象和权限。

165. 一种多媒体文件，包括：  
至少一个编码视频轨；  
至少一个编码音频轨；

多个编码文本串；

其中所述编码文本串描述了至少一个视频轨和至少一个音频轨的特征。

166. 根据权利要求 165 所述的多媒体文件，其中多个文本串使用不同的语言描述了视频轨或音频轨的同一特征。

167. 根据权利要求 166 所述的多媒体文件，进一步包括：

至少一个编码字幕轨；

其中所述多个编码文本串包括描述所述字幕轨特征的串。

168. 一种用于创建多媒体文件的系统，其包括：

处理器，其被配置成：

编码至少一个视频轨；

编码至少一个音频轨；

将所述编码音频轨的至少一个与视频轨交错；

插入文本串，其中所述文本串以多种语言来描述所述至少一个视频轨和所述至少一个音频轨的大量特征中的每一个。

167. 一种用于显示多媒体文件的系统，该多媒体文件包含编码的音频、视频和文本串，所述系统包括：

处理器，其被配置成：

从所述文件中提取所述编码文本串；以及

使用所述文本串来产生下拉菜单显示。



## 多媒体分发系统

### 技术领域

【0001】本发明一般涉及多媒体文件的编码、传输和解码。更具体地，本发明涉及能够包括单一音频轨和单一视频轨之外磁轨的多媒体文件的编码、传输和解码。

### 背景技术

【0002】互连网的发展刺激了用于多媒体信息的文件格式的发展，以使得文件能够标准化的产生、分发和显示。典型地，单一多媒体文件包括单一的视频轨和单一的音频轨。当多媒体被写入诸如数字视频光盘（DVD）的高容量和物理移动式媒体时，能够使用多个文件，使用上述文件以提供大量的视频轨、音频轨和字幕轨。能够提供额外的文件，其包含着能够用于产生交互菜单的信息。

### 发明内容

【0003】本发明的实施例包括用于产生、分发、和解码多媒体文件的多媒体文件和系统。在本发明的一个方面中，多媒体文件包括多个编码视频轨。在本发明的另一个方面中，多媒体文件包括多个编码音频轨。在本发明的另一个方面中，多媒体文件包括至少一个字幕轨。在本发明的另一个方面中，多媒体文件包括编码的“元数据（meta data）”。在本发明的另一个方面中，多媒体文件包括编码菜单信息。

【0004】根据本发明实施例的多媒体文件包括多个编码视频轨。在本发明进一步的实施例中，多媒体文件包括多个连接“RIFF(资源交换文件格式)”块，并且每个编码视频轨被包含在一个分离的“RIFF”块中。此外，使用心理视觉增强来编码该视频，并且每个视频轨至少有一个与之关联的音频轨。

【0005】在另一个实施例中，每个视频轨编码为“RIFF”块之内的一

系列“视频 (video)”块，并且伴随每个视频轨的音频轨被编码为一串“音频 (audio)”块，所述的“音频”块在包含关联视频轨的“视频”块的“RIFF”块内交错。而且，每个“视频”块能够包含可用于从视频轨产生单一的视频帧的信息，以及每个“音频”块包含来自音频轨一部分的音频信息，该音频轨的一部分伴随着使用“视频”块所产生的帧。此外，“音频”块能够被交错在“RIFF”块内的相应“视频”块之前。

**【0006】**根据本发明的编码多媒体文件的系统包括一处理器，其被配置成编码多个视频轨、连接编码视频轨，并将连接的编码视频轨写入单一文件。在另一个实施例中，处理器被配置成编码所述视频轨，以致每个视频轨被包含在分离的“RIFF”块中，并且所述处理器被配置成利用心理视觉增强来编码该视频。此外，每个视频轨至少有一个与之关联的音频轨。

**【0007】**在另一个实施例中，处理器被配置成将每个视频轨编码为一系列在“RIFF”块中的“视频”块，并且至少一个伴随每个视频轨的音频轨被编码为一系列“音频”块，所述的“音频”块在包含关联视频轨的“视频”块的“RIFF”块内交错。

**【0008】**在另一个进一步的实施例中，处理器被配置成编码所述视频轨，以致每个“视频”块包含能够被用于从视频轨来产生单一的视频帧的信息，而且处理器被配置成编码与视频轨关联的音频轨，以致每个“音频”块包含来自音频轨一部分的音频信息，该音频轨的一部分伴随着使用“视频”块所产生的帧。此外，处理器能够被配置成将每个“音频”块交错在“RIFF”块内的相应“视频”块之前。

**【0009】**根据本发明实施例的用于解码包含多个编码视频轨的多媒体文件的系统包括处理器，该处理器被配置成从多媒体文件中提取信息。配置该处理器以提取包含在多媒体文件中的关于编码视频轨数目的信息。

**【0010】**在一个进一步的实施例中，配置该处理器以将编码视频轨设置在“RIFF”块中。此外，第一编码视频轨能够包含在一个具有标准4cc码的第一“RIFF”块中，第二视频轨能够包含在具有专用4cc码的

第二“RIFF”块中，并且所述专用 4cc 码能够以标准 4cc 码的前两个字符作为它的最后两个字符。

**【0011】** 在一个附加的实施例中，每个编码视频轨被包含在一个分离的“RIFF”块中。

**【0012】** 在另一个进一步的实施例中，解码视频轨类似于在多媒体文件创建时被编码的原始视频轨，而且解码视频轨和原始视频轨之间差异中的至少一些被设置在视频轨各帧的暗部分。更进一步地，解码视频轨和原始视频轨之间的差异能够被设置在视频轨的高速运动场景中。

**【0013】** 在一个附加的实施例中，每个视频轨具有至少一个与它关联的音频轨。

**【0014】** 在一个进一步的附加实施例中，处理器被配置成通过解码一系列的在“RIFF”块之内的“视频”块，来显示来自视频轨的视频，并且通过解码在“RIFF”块中交错的一系列的“音频”块，来产生来自伴随视频轨的音频轨的音频，所述的“RIFF”块包含关联视频轨的“视频”块。

**【0015】** 在一个更进一步的实施例中，处理器被配置成使用提取自每个“视频”块的信息来产生视频轨的一个单一帧，使用提取自每个“音频”块的信息来产生音频轨的一部分，其伴随着使用“视频”块而产生的帧。此外，处理器能够被配置成在“视频”块之前布置“音频”块，该“音频”块和该“视频”块在“RIFF”块中相关联。

**【0016】** 根据本发明实施例的多媒体文件包括一系列编码视频帧和在编码视频帧之间交错的编码音频帧。该编码音频包括两个或更多个轨的音频信息。

**【0017】** 在一个进一步的实施例中，至少一个音频信息轨包括多个音频通道或声道。

**【0018】** 另一个实施例进一步包括标识了包含在多媒体文件中的音频轨的数目的头信息和关于至少一个音频信息轨的描述信息。

**【0019】** 在一个进一步的实施例中，每个编码视频帧之前有编码音频

信息，并且在视频帧之前的编码音频信息包括每个音频轨中伴随着编码视频帧的那部分的音频信息。

**【0020】**又在另一个实施例中，视频信息在多媒体文件中存储为块。此外，每个视频信息块能够包括单一视频帧。而且，音频信息能够在多媒体文件中被存储为块，并且来自两个分离的音频轨的音频信息不包含在单一的音频信息块中。

**【0021】**在一个更进一步的实施例中，“视频”块依靠至少一个“音频”块而与每个音频轨分离，并且分离“视频”块的“音频”块包含着伴随视频信息的音频轨的一部分的音频信息，所述视频信息包含在该“音频”块之后的“视频”块之中。

**【0022】**根据本发明实施例的编码多媒体文件的系统包括处理器，其被配置成编码视频轨、编码多个音频轨、将来自视频轨的信息与来自多个音频轨的信息交错，以及将交错后的视频和音频信息写入单一文件中。

**【0023】**在一个进一步的实施例中，至少一个音频轨包含了多个音频通道。

**【0024】**在另一个实施例中，处理器被进一步配置成对标识编码音频轨数目的头信息进行编码，并将此头信息写入单一文件中。

**【0025】**在一个进一步的实施例中，处理器被进一步配置成对标识关于至少一个编码音频轨的描述信息的头信息进行编码，并将此头信息写入单一文件中。

**【0026】**又在另一个实施例中，处理器编码视频轨为视频信息块。此外，处理器能够将每个音频轨编码为一系列的音频信息块。而且，每个音频信息块能够包含来自单一的音频轨的音频信息，并且处理器能够被配置成在视频信息块之间交错音频信息块。

**【0027】**在一个更进一步的实施例中，处理器被配置成编码每个音频轨的一部分，其伴随着“音频”块中的“视频”块的视频信息，并且配置处理器，使“视频”块和“音频”块被交错成：使包含来自每个音频轨的音频信息的“音频”块处于每个“视频”块之前，所述每个

音频轨伴随着包含在“视频”块中的视频信息。此外，处理器能够被配置成编码所述视频轨，以致一单一的视频帧能够包含在每个“视频”块之内。

**【0028】** 又在一个更进一步的实施例中，处理器是通用处理器。

**【0029】** 在一个附加的进一步的实施例中，处理器是专用电路。

**【0030】** 根据本发明的、用于解码包含多个音频轨的多媒体文件的系统包括处理器，其被配置成从多媒体文件中提取信息。配置该处理器以提取包含在多媒体文件中的关于音频轨数目的信息。

**【0031】** 在一个进一步的实施例中，处理器被配置成从多个音频轨中选择一单一的音频轨，以及配置该处理器以从所选的音频轨中解码音频信息。

**【0032】** 在另一实施例中，至少一个音频轨包括多个音频通道。

**【0033】** 在一个更进一步的实施例中，处理器被配置成从包含描述信息的多媒体文件头中提取信息，该描述信息关于至少一个音频轨。

**【0034】** 根据本发明实施例的、用于传递多媒体信息的系统包括网络、包含多媒体文件并通过服务器连接到所述网络的存储装置，以及连接到网络的客户机。客户机能够请求来自服务器的多媒体文件传送，并且该多媒体文件包括至少一个视频轨和多个伴随视频轨的音频轨。

**【0035】** 根据本发明的多媒体文件包括一系列编码视频帧和至少一个在编码视频帧之间交错的编码字幕轨。

**【0036】** 在一个进一步的实施例中，所述至少一个编码字幕轨包括多个编码字幕轨。

**【0037】** 另一个实施例进一步包括头信息，其标识了在多媒体文件中所包含的编码字幕轨的数目。

**【0038】** 一个更进一步的实施例还包括头信息，其包括关于至少一个编码字幕轨的描述信息。

**【0039】** 在又一个实施例中，每个字幕轨包括一系列位图，每个字幕轨可包括一系列被压缩位图。此外，可使用游程长度编码来压缩每个

位图。

**【0040】** 在一个更进一步的实施例中，所述一系列编码视频帧被编码为一系列的视频块，每个编码字幕轨被编码为一系列字幕块。每个字幕块包含信息，其能够作为文本在显示器上显示。此外，每个字幕块能够包含关于一单一字幕的信息。而且，每个字幕块可包括关于视频序列的一部分的信息，字幕将被叠加在该视频序列的那一部分之上。

**【0041】** 在又一个实施例中，每个字幕块包括关于显示器的一部分的信息，字幕将位于显示器的该部分之上。

**【0042】** 又在一个进一步的实施例中，每个字幕块包括关于字幕颜色的信息，并且该关于颜色的信息可包括一调色板。此外，字幕块可包括包含关于第一调色板的信息的第一字幕块和包含关于第二调色板的信息的第二字幕块，其中关于第二调色板的信息接替了关于第一调色板的信息。

**【0043】** 根据本发明实施例的、用于编码多媒体文件的系统包括处理器，其被配置成编码视频轨、编码至少一个字幕轨、将来自视频轨的信息和来自至少一个字幕轨的信息相交错，以及将交错后的视频和字幕信息写入单一文件。

**【0044】** 在一个进一步的实施例中，所述至少一个字幕轨包括多个字幕轨。

**【0045】** 在另一个实施例中，处理器被进一步配置成将标识多媒体文件中所包含的字幕轨数目的头信息编码，并写入单一文件。

**【0046】** 又在一个进一步的实施例中，处理器被进一步配置成将关于至少一个字幕轨的描述信息编码，并写入单一文件。

**【0047】** 在另一个进一步的实施例中，视频轨被编码为视频块，而至少一个字幕轨的每一个被编码为字幕块。此外，每个字幕块能够包含伴随视频轨一部分的单一的字幕，并且可配置交错器（interleaver）以在包含视频轨一部分的视频块之前交错每个字幕块，其中字幕块之内的字幕伴随着所述包含视频轨一部分的视频块。

**【0048】** 在一个更进一步的实施例中，处理器被配置成通过将字幕编

码为位图来产生字幕块。

**【0049】**在又一个实施例中，字幕被编码为压缩位图。此外，可使用游程长度编码来压缩位图。而且，处理器可在每个字幕块中包括信息，该信息关于视频序列的一部分，在该部分上字幕将被叠加。

**【0050】**在又一个进一步的实施例中，处理器在每个字幕块中包括信息，该信息关于显示器的一部分，字幕将位于该部分上。

**【0051】**在又一个实施例中，处理器在每个字幕块中包括关于字幕颜色的信息。

**【0052】**在一个更进一步的实施例中，关于颜色的信息包括调色板。此外，字幕块可包括包含关于第一调色板的信息的第一字幕块和包含关于第二调色板的信息的第二字幕块，所述关于第二调色板的信息接替了关于第一调色板的信息。

**【0053】**根据本发明的实施例的、解码多媒体文件的系统包括处理器，其被配置成从多媒体文件中提取信息。所述处理器被配置成检查多媒体文件，从而确定是否有至少一个字幕轨存在。此外，至少一个字幕轨可包括多个字幕轨，并且所述处理器被配置成确定多媒体文件中的字幕轨的数目。

**【0054】**在一个进一步的实施例中，处理器被进一步配置成提取头信息，其标识了来自多媒体文件字幕轨的数量。

**【0055】**在另一个实施例中，处理器被进一步配置成提取关于来自多媒体文件的至少一个字幕轨的描述信息。

**【0056】**又在另一个进一步的实施例中，多媒体文件包括至少一个被编码为视频块的视频轨，以及多媒体文件包括至少一个编码为字幕块的字幕轨。

**【0057】**又在另一个实施例中，每个字幕块包括关于单一字幕的信息。

**【0058】**在一个更进一步的实施例中，每个字幕在字幕块中编码为位图，处理器被配置成解码视频轨，以及处理器被配置成通过将位图叠加在视频序列的一部分上，而构造用于显示的视频帧。此外，可将字幕编

码为压缩位图，并且可配置处理器以解压该位图。而且，可配置处理器以解压经游程长度编码后的位图。

**【0059】**在又一个实施例中，每个字幕块包括关于视频轨一部分的信息，在该部分上字幕将被叠加，以及配置处理器以通过将字幕的位图叠加在每个视频帧之上，来产生用于显示的视频帧序列，其中所述的每个视频帧由字幕块中的信息来表示。

**【0060】**在一个附加的进一步的实施例中，每个字幕块包括关于帧内位置的信息，字幕将位于该信息所指示的位置，以及配置处理器以在每个视频帧内的、由字幕块之内的信息所指示的位置之上叠加字幕。

**【0061】**在另一个附加的实施例中，每个字幕块包括关于字幕颜色的信息，并且处理器被配置成将字幕块之内所指示的颜色（一种或多种）叠加在字幕上。此外，字幕块之内的颜色信息可包括调色板，并配置处理器以使用调色板叠加字幕，获得字幕位图中所使用的颜色。而且，字幕块可包括包含关于第一调色板的信息的第一字幕块和包含关于第二调色板的信息的第二字幕块，并且处理器可被配置成使用第一调色板叠加到字幕来获得信息，该信息关于第一块被处理之后字幕位图所使用的颜色的信息，以及处理器可被配置成使用第二调色板叠加到字幕来获得信息，该信息关于第二块被处理之后字幕位图所使用的颜色的信息。

**【0062】**根据本发明实施例的、用于传递多媒体信息的系统包括网络、包含多媒体文件并通过服务器连接到网络的存储装置，以及连接到网络的客户机。客户机能够请求来自服务器的多媒体文件传送，并且该多媒体文件包括至少一个视频轨和至少一个伴随视频轨的字幕轨。

**【0063】**根据本发明实施例的多媒体文件包括一系列编码视频帧和编码菜单信息。此外，编码菜单信息可作为块来存储。

**【0064】**一个进一步的实施例还包括至少两个分离的菜单信息的“Menu（菜单）”块，并且至少两个分离的“Menu”块能够包含在至少两个分离的“RIFF”块中。

**【0065】**在另一个实施例中，包含“Menu”块的第一“RIFF”块包括



标准 4cc 码，以及包含“Menu”块的第二“RIFF”块包括专用 4cc 码，在其中标准 4cc 码的前两个字符作为专用 4cc 码的最后两个字符出现。

【0066】又在一个进一步的实施例中，至少两个分离的“Menu”块被包含在一单一的“RIFF”块中。

【0067】又在另一个实施例中，“Menu”块包括描述一系列菜单的块和包含与该系列菜单关联的媒体的“MRIF”块。此外，该“MRIF”块能够包含媒体信息，该媒体信息包括了视频轨、音频轨和覆盖轨。

【0068】在一个更进一步的实施例中，描述一系列菜单的块能够包含描述全部菜单系统的块、至少一个按语言来分组菜单的块、至少一个描述个体菜单显示并伴随背景音频的块、至少一个描述菜单上按钮的块、至少一个描述按钮在显示器上位置的块，以及至少一个描述与按钮关联的各种动作的块。

【0069】又一个实施例还包括到第二文件的链接。编码菜单信息包含在第二文件之内。

【0070】根据本发明实施例的、用于编码多媒体文件的信息包括处理器，其被配置成编码菜单信息。处理器还可被配置成产生包括编码视频轨和编码菜单信息的多媒体文件。此外，处理器可被配置成产生菜单的对象模型，将对象模型转换为配置文件，将该配置文件语法分析为块，产生包含媒体信息的 AVI 文件，将 AVI 文件中的媒体交错到“MRIF”块，并将所解析的块与“MRIF”块连接，以创建“Menu”块。而且，处理器可被进一步配置成使用对象模型来产生第二较小“Menu”块。

【0071】在一个进一步的实施例中，处理器被配置成编码第二菜单，该处理器可将第一编码菜单插入到第一“RIFF”块中，以及将第二编码菜单插入到第二“RIFF”块中。

【0072】在另一个实施例中，处理器在一单一的“RIFF”块中包含第一和第二编码菜单。

【0073】又在一个进一步的实施例中，处理器被配置成将第二文件中编码菜单的引用插入到多媒体文件。

【0074】根据本发明实施例的、用于解码多媒体文件的系统包括处理器，其被配置成从多媒体文件来提取信息。配置该处理器以检查多媒体文件，从而确定是否包含了编码菜单信息。此外，可配置处理器以从“RIFF”块之内的“Menu”块提取菜单信息，以及配置处理器以使用存储在“Menu”块中的视频信息来构造菜单显示。

【0075】在一个进一步的实施例中，处理器被配置成产生伴随菜单显示的背景音频，其使用了存储在“Menu”块中的音频信息。

【0076】在另一个实施例中，处理器被配置成通过将来自“Menu”块的覆盖物（overlay）覆盖到来自“Menu”块的视频信息之上，来产生菜单显示。

【0077】根据本发明实施例的、用于传递多媒体信息的系统包括网络、包含多媒体文件并通过服务器连接到网络的存储装置，以及连接到网络的客户机。客户机能够请求来自服务器的多媒体文件传送，并且该多媒体文件包括编码的菜单信息。

【0078】多媒体文件包括一系列编码视频帧和有关多媒体文件的编码的元数据。编码的元数据包括至少一个语句，其包括主题、谓词、对象和权限。此外，主题能够包含标识由元数据来描述的文件、项目、人员或组织的信息，谓词能够包含指示主题特征的信息，对象能够包含由谓词所指示的描述主题特征的信息，以及权限能够包含关于所述语句的源的信息。

【0079】在一个进一步的实施例中，主题是包括了类型和数值的块，在其中，数值包含信息以及类型指示块是资源还是匿名节点或网点（node）。

【0080】在另一个实施例中，谓词是一包括类型和数值的块，在其中，数值包含信息以及类型指示数值信息是谓词 URI 还是顺序列表条目。

【0081】又在一个进一步的实施例中，对象是一包括类型、语言、数据类型和数值的块，在其中，数值包含信息，类型指示该数值信息是 UTF-8 文字，还是整数文字或整数字面（literal integer），或是文字 XML 数据，数据类型指示数值信息的类型，以及语言包含标识具体语言的

信息。

【0082】又在另一个实施例中，权限是一包括类型和数值的块，在其中，数值包含信息，以及类型指示数值信息是语句的权限。

【0083】在一个更进一步的实施例中，编码的数据的至少一部分表示为二进制数据。

【0084】在更进一步的实施例中，编码的数据的至少一部分表现为 64 位 ASCII 数据。

【0085】在一个更进一步的实施例中，编码的数据的至少第一部分表现为二进制数据，以及编码的数据的至少第二部分表现为包含表现为第二种格式的数据的附加块。此外，每个附加块能够包含单段元数据。

【0086】根据本发明的实施例的、编码多媒体文件的系统包括处理器，其被配置成编码视频轨。还可配置处理器以编码关于多媒体文件的元数据，以及编码的元数据包括至少一个语句，其包含了主题、谓词、对象和权限。此外，主题能够包含标识由元数据来描述的文件、项目、人员或组织的信息，谓词可包括指示主题特征的信息，对象可包括由谓词所指示的描述主题特征的信息，以及权限可包括关于语句源的信息。

【0087】在一个进一步的实施例中，处理器被配置成将主题编码为块，该块包括了类型和数值，在其中，数值包含信息以及类型指示块是资源，还是匿名节点。

【0088】在另一个实施例中，处理器被配置成将谓词编码为块，该块包括类型和数值，在其中，数值包含信息以及类型指示数值信息是谓词 URI 还是顺序列表条目。

【0089】又在一个进一步的实施例中，处理器被配置成将对象编码为块，该块包括类型、语言、数据类型和数值，在其中，数值包含信息，类型指示该数值信息是 UTF-8 文字，还是整数文字或整数字面，或是文字 XML 数据，数据类型指示数值信息的类型，以及语言包含标识具体语言的信息。

【0090】又在另一个实施例中，处理器被配置成将权限编码为块，该

块包括类型和数值的块，在其中，数值包含信息，以及类型指示数值信息是语句的权限。

**【0091】** 在一个更进一步的实施例，处理器被进一步配置成将关于多媒体文件的元数据的至少一部分编码为二进制数据。

**【0092】** 在又一个实施例中，处理器被进一步配置成将关于多媒体文件的元数据的至少一部分编码为 64 位 ASCII 数据。

**【0093】** 在一个附加的实施例中，处理器被进一步配置成将关于多媒体文件的元数据的至少第一部分编码为二进制数据，以及将关于多媒体文件的元数据的至少第二部分编码为附加块，该附加块包含表现为第二种格式的数据。此外，处理器可被配置成编码包含单段元数据的附加块。

**【0094】** 根据本发明的、用于解码多媒体文件的系统包括处理器，其被配置成从多媒体文件提取信息。配置处理器以提取关于多媒体文件的元数据信息，该元数据信息包括至少一个语句，该语句包括主题、谓词、对象和权限。此外，处理器可被配置成从主题中提取信息，该信息标识了由元数据所描述的文件、项目、人员或组织。而且，可配置处理器以从谓词中提取指示主题特征的信息，可配置处理器以从对象中提取描述所述谓词所指示的主题特征的信息，以及可配置处理器以从权限中提取关于语句源的信息。

**【0095】** 在一个进一步的实施例中，主题是一个包括了类型和数值的块，并且所述处理器被配置成通过检查类型来识别该块是否包含主题信息，而且所述处理器被配置成从数值来提取信息。

**【0096】** 在另一个实施例中，谓词是一个包括类型和数值的块，并且所述处理器被配置成通过检查类型来识别该块是否包含谓词信息，以及所述处理器被配置成从数值中提取信息。

**【0097】** 又在一个进一步的实施例中，对象是一个包括类型、语言、数据类型和数值的块，并且所述处理器被配置成通过检查类型来识别该块是否包含对象信息，处理器被配置成检查数据类型以确定包含在数值中的信息的数据类型，处理器被配置成从数值中提取由数据类型

所指示的类型的信息，以及所述处理器被配置成从语言中提取出标识具体语言的信息。

**【0098】**又在另一个实施例中，权限是一个块，该块包括类型和数值的块，处理器被配置成通过检查类型来识别该块是否包含权限信息，以及所述处理器被配置成从数值提取信息。

**【0099】**在一个更进一步的实施例中，处理器被配置成从元数据语句中提取信息，并显示信息的至少一部分。

**【0100】**在又一个实施例中，处理器被配置成使用元数据来构建指示内存中定向标签图形（directed-labeled graph）的数据结构。

**【0101】**在一个更进一步的实施例中，处理器被配置成通过检查多个语句的主题、谓词、对象和权限中的至少之一，来从头到尾地搜索元数据以获得信息。

**【0102】**在又一个实施例中，处理器被配置成显示作为图形用户界面的一部分的搜索结果。此外，处理器可被配置成响应外部设备的请求，执行搜索。

**【0103】**在一个附加的进一步的实施例中，关于多媒体文件的元数据信息的至少一部分表现为二进制数据。

**【0104】**在另一个附加的实施例中，关于多媒体文件的元数据信息的至少一部分表现为 64 位 ASCII 数据。

**【0105】**在另一个进一步的实施例中，关于多媒体文件的元数据信息的至少第一部分表现为二进制数据，以及关于多媒体文件的元数据信息的至少第二部分表现为附加块，该块包含了表现为第二种格式的数据。此外，附加块能够包含单段元数据。

**【0106】**根据本发明的、用于传递多媒体信息的系统包括网络、包含多媒体文件并通过服务器连接到所述网络的存储装置，以及连接到网络的客户机。客户机能够请求来自服务器的多媒体文件传送，并且该多媒体文件包括关于多媒体文件的元数据，以及该元数据包括至少一个语句，该语句包括主题、谓词、对象以及权限。

【0107】根据本发明的多媒体文件包括至少一个编码视频轨、至少一个编码音频轨，以及多个编码的文本串。该编码的文本串描述了至少一个视频轨和至少一个音频轨的特征。

【0108】在一个进一步的实施例中，多个文本串使用不同的语言描述了视频轨或音频轨的相同特征。

【0109】另一个实施例还包括至少一个编码字幕轨。多个编码的文本串包括描述字幕轨特征的串。

【0110】根据本发明的、用于创建多媒体文件的系统包括处理器，其被配置成编码至少一个视频轨、编码至少一个音频轨、将至少一个编码音频轨和视频轨交错，以及插入文本串，其中所述的文本串以多种语言来描述至少一个所述视频轨和至少一个所述音频轨的许多特征中的每一个。

【0111】根据本发明的、用于显示诸如编码音频、视频和文本串的多媒体文件的系统包括一个处理器，其被配置成从文件提取编码的文本串，并使用文本串产生一下拉菜单显示。

## 附图说明

【0112】图 1 是根据本发明一个实施例的系统的示意图，该系统用于编码、分发和解码文件。

【0113】图 2.0 是根据本发明一个实施例的多媒体文件结构的示意图。

【0114】图 2.0.1 是根据本发明另一实施例的多媒体文件结构的示意图。

【0115】图 2.1 是根据本发明一个实施例的“hdr1”列表块的概念图。

【0116】图 2.2 是根据本发明一个实施例的“str1”块的概念图。

【0117】图 2.3 是根据本发明一个实施例的内存的概念图，该内存被分配以存储多媒体文件的“DXDT”块。

【0118】图 2.3.1 是“元数据”块的概念图，所述的“元数据”块能够包含在根据本发明实施例的多媒体文件的“DXDT”块中。

【0119】图 2.4 是根据本发明一个实施例的“DMNU”块的概念图。

【0120】图 2.5 是菜单块的概念图,所述菜单块被包含在根据本发明一个实施例的“WowMenuManager”块中。

【0121】图 2.6 是菜单块的概念图,所述菜单块被包含在根据本发明另一实施例的“WowMenuManager”块中。

【0122】图 2.6.1 是概念图,其示出了包含在“DMNU”块之内的多个块之间的关联。

【0123】图 2.7 是根据本发明一个实施例的、多媒体文件的“流数据(movi)”列表块的概念图。

【0124】图 2.8 是根据包含 DRM 的本发明一个实施例的、多媒体文件的“movi”列表块的概念图。

【0125】图 2.9 是根据本发明一个实施例的、“DRM”块的概念图。

【0126】图 3.0 是根据本发明一个实施例的、用于产生多媒体文件的系统的框图。

【0127】图 3.1 是根据本发明一个实施例来产生“DXDT”块的系统的框图。

【0128】图 3.2 是根据本发明一个实施例来产生“DMNU”块的系统的框图。

【0129】图 3.3 是根据本发明一个实施例的媒体模型的概念图。

【0130】图 3.3.1 是来自媒体模型的对象的概念图,所述媒体模型能够用来自动产生根据本发明一个实施例的小菜单。

【0131】图 3.4 是一处理过程的流程图,该处理过程被用来根据本发明一个实施例来再成块(re-chunk)音频。

【0132】图 3.5 是根据本发明一个实施例的视频编码器的框图。

【0133】图 3.6 是一方法的流程图,该方法用来执行根据本发明实施例的、对 I 帧的平滑心理视觉增强。

【0134】图 3.7 是一处理过程的流程图，该处理过程用于执行根据本发明一个实施例的、宏块 SAD 心理视觉增强。

【0135】图 3.8 是一处理过程的流程图，该处理过程用于根据本发明一个实施例的一遍速率控制。

【0136】图 3.9 是一处理过程的流程图，该处理过程用于执行根据本发明一个实施例的、第 N 遍 VBV 速率控制。

【0137】图 4.0 是一处理过程的流程图，该处理过程根据本发明的实施例，来设置所需的来自多媒体文件的多媒体信息，并显示该多媒体信息。

【0138】图 4.1 是根据本发明一个实施例的解码器的框图。

【0139】图 4.2 是根据本发明一个实施例所显示的菜单的实例。

【0140】图 4.3 是一概念图，其显示了根据本发明一个实施例的、用来产生示于图 4.2 的显示的信息源。

## 具体实施方式

【0141】参考附图，本发明的实施例能够编码、传输和解码多媒体文件。根据本发明实施例的多媒体文件能够包含多媒体视频轨、多媒体音频轨、多媒体字幕轨、能够被用于产生用来访问文件内容的菜单界面的数据，以及关于文件内容的“元数据”。根据本发明多个实施例的多媒体文件还包括文件之外的视频轨、音频轨、字幕轨和“元数据”的引用。

### 1. 系统描述

【0142】现转到图 1，显示了根据本发明实施例的用于编码、分发和解码文件的系统。系统 10 包括计算机 12，其通过网络 14 被连接到多个其它计算装置。可被连接到网络的装置包括服务器 16、膝上型计算机 18，以及个人数码助理（PDA）20。在多个实施例中，在所述装置和网络之间的连接可以是有线的或无线的，并可使用多种联网协议来实现。



【0143】在操作中，计算机 12 可被用来编码根据本发明实施例的多媒体文件。计算机 12 还能够用来解码根据本发明实施例的多媒体文件和分发根据本发明实施例的多媒体文件。计算机可使用多种包括通过对等网络的传送协议或传输协议，来分发文件。此外，计算机 12 能够将根据本发明实施例的多媒体文件传送到服务器 18，在其该服务器中，其它装置访问可该文件。其它装置可包括任意多个计算装置或甚至是专用解码器装置。在所示的实施例中，显示了膝上型计算机和 PDA。在其它实施例中，数字机顶盒、台式计算机、游戏机、消费类电子装置以及其它装置可连接到网络，下载并解码多媒体文件。

【0144】在一个实施例中，装置通过网络从服务器访问多媒体文件。在其它实施例中，装置通过对等网络从许多计算机访问多媒体文件。在多个实施例中，多媒体文件可被写入便携式存储装置，例如磁盘驱动器、CD-ROM 或 DVD。在许多实施例中，电子装置可访问写入便携式存储装置的多媒体文件。

## 2. 文件结构描述

【0145】可构建根据本发明实施例的多媒体文件以兼容资源交换文件格式(“RIFF 文件格式”)，所述资源交换文件格式由位于 Redmond, WA 的微软公司 (Microsoft Corporation) 和位于 Armonk, NY 的 IBM 公司 (International Business Machines Corporation ) 定义。RIFF 是一种用于存储多媒体数据及其关联信息的文件格式。RIFF 文件通常具有 8 字节 RIFF 头，其标识了文件并提供了该文件头之后的文件剩余长度 (即 `file_length - 8`)。RIFF 文件的全部剩余包括了“块”和“列表”。每个块具有 8 字节的块头，其标识了块的类型并以字节为单位给出了跟随该块头的数据的长度。每个列表具有 8 字节列表头，其标识了列表的类型并以字节为单位给出了跟随该列表头的数据的长度。列表中的数据包括块和/或其它列表 (其接下来可包括块和/或其它列表)。RIFF 列表有时还被称为“列表块”。

【0146】AVI 文件是一种特定形式的 RIFF 文件，AVI 遵循 RIFF 文件的格式，但包括了多种具有定义了的标识符的块和列表，其以特定格

式包含着多媒体数据。Microsoft Corporation 开发和定义了该 AVI 格式。通常使用能够以 AVI 格式输出多媒体数据的解码器来创建 AVI 文件。通常依靠总体上公知为 AVI 解码器的一组软件，来解码 AVI 文件。

**【0147】** RIFF 和 AVI 格式是灵活的，这在于它们仅仅定义了块和列表，块和列表是所定义的文件格式的部分，除此以外，RIFF 和 AVI 格式还允许文件包含在 RIFF 和/或 AVI 文件格式定义以外的列表和/或块，并且不翻译对于 RIFF 和/或 AVI 解码器不可读的文件。实际上，可实现 AVI（以及类似地 RIFF）解码器，它们完全忽略那些包含没有在 AVI 文件格式定义中找到的头信息的列表和块。该 AVI 解码器一定还会从头到尾读这些非 AVI 的块和列表，以致于 AVI 解码器的操作减慢，但另外这些非 AVI 的块和列表一般对于 AVI 解码器起不到作用，从而被 AVI 解码器所忽略。

**【0148】** 在图 2.0 中描述了根据本发明实施例的多媒体文件。所描述的多媒体文件 30 包括字符集块（“CSET”块）32，信息列表块（“INFO”列表块）34，文件头块（“hdrl”列表块）36，元数据块（“DXDT”块）38，菜单块（“DMNU”块）40，废弃块（“junk”块）41，电影列表块（“movi”列表块）42，可选索引块（“idxl”块）44 以及第二菜单块（“DMNU”块）46。这些块中的许多和其它块中的部分被定义在 AVI 文件格式中，而其它的块没有包含在 AVI 文件格式中。在许多但不是全部情况下，下面的讨论标识了块或块的部分，其定义为 AVI 文件格式的一部分。

**【0149】** 在图 2.0.1 中描述了根据本发明实施例的另一个多媒体文件。该多媒体文件 30' 类似于示于图 2.0 的多媒体文件，除了该文件包括多个连接“RIFF”块。该“RIFF”块可包括类似于示于图 2.0 的“RIFF”块，该“RIFF”块可把第二“DMNU”块 46 排除在外或可以“DMNU”块 46' 的形式包含菜单信息。

**【0150】** 在所描述的实施例中，多媒体包括多个连接“RIFF”块，在其中第一“RIFF”块 50 包括字符集块（“CSET”块）32'，信息列表块（“INFO”列表块）34'，文件头块（“hdrl”列表块）36'，元数据块（“DXDT”块）38'，菜单块（“DMNU”块）40'，废弃块（“junk”块）41'，电影

列表块 (“movi” 列表块) 42’，以及选择性的索引块 (“idx1” 块) 44’。第二 “RIFF” 块 52 包含了第二菜单块 (“DMNU” 块) 46’。包含附加标题的附加 “RIFF” 块 54 可被包含在 “RIFF” 菜单块 52 之后。附加的 “RIFF” 块能够包含兼容 AVI 文件格式的独立的媒体。在一个实施例中，第二菜单块 46’和附加的 “RIFF” 块具有专用的 4 个字符码（以 AVI 格式定义，这将在后文讨论），所述 4 个字符码的第一两字符作为第二两字符出现，所述 4 个字符码的第二两字符作为第一两字符出现。

## 2.1 “CSET” 块

【0151】“CSET” 块 32 是一种以 Microsoft Corporation 创建的音频视频交错文件格式 (AVI 文件格式) 定义的块。“CSET” 块定义了多媒体文件的字符集和语言信息。包含根据本发明实施例的 “CSET” 块是可选的。

【0152】根据本发明一个实施例的多媒体文件并没有采用 “CSET” 块，而是采用了 Unicode Consortium 定义的 UTF-8，用于默认与 RFC 3066 语言规范相结合的字符集，RFC 3066 语言规范由 Internet Engineering Task Force 为语言信息而定义。

## 2.2 “INFO” 列表块

【0153】“INFO” 列表块 34 能够存储帮助标识多媒体文件内容的信息。“INFO” 列表是以 AVI 文件格式定义的，并且在根据本发明实施例的多媒体文件中是否包含 “INFO” 列表是可选的。许多包含 “DXDT” 块的实施例并不包括 “INFO” 列表块。

## 2.3 “hdrl” 列表块

【0154】“hdrl” 列表块 38 是以 AVI 文件格式定义的，并且提供了关于多媒体文件中数据类型的信息。通常需要包含 “hdrl” 列表块或包含类似描述信息的块。“hdrl” 列表块包含用于每个视频轨、每个音频轨和每个字幕轨的块。

【0155】图 2.1 示出了根据本发明一个实施例的 “hdrl” 列表块 38 的概念图，其包含了一个单一的视频轨 62、两个音频轨 64、外部音频轨 66、

两个字幕轨 68，以及外部字幕轨 70。该“hdrl”列表 60 包括“avih”块。“avih”块 60 包含整个文件的全局信息，例如文件之内的流的数目和包含在多媒体文件中的视频的宽度和高度。“avih”块可根据 AVI 文件格式来实现。

**【0156】**除了“avih”块之外，“hdrl”列表也包括有用于每个音频、视频和字幕轨的流描述符列表。在一个实施例中，是利用“strl”块来实现该流描述符列表的。在图 2.2 中示出了根据本发明实施例的“strl”块。每个“strl”块用来描述多媒体文件中的每个轨。用于多媒体文件中音频、视频和字幕轨的“strl”块包括一“strl”块，其引用了“strh”块 92、“strf”块 94、“strd”块 96 以及“strn”块 98。所有这些块可根据 AVI 文件格式来实现。特别感兴趣的是指定了媒体轨类型的“strh”块 92 和可被修改以指示视频是否依靠数字版权管理来保护的“strd”块 96。下文提供了根据本发明实施例的数字版权管理的多种实现的讨论。

**【0157】**根据本发明实施例的多媒体文件能够包含对外部文件的引用，所述外部文件保存了多媒体信息，例如附加的音频轨或附加的字幕轨。这些轨的引用可被包含在“hdrl”块或废弃块 41 中。在这两种情况的任何一个中，引用能够包含在“strl”块 90 的“strh”块 92 中，其引用本地文件或远处存储的文件。被引用的文件可以是标准 AVI 文件或者是包含了附加轨的根据本发明实施例的多媒体文件。

**【0158】**在附加实施例中，被引用文件能够包含出现在引用文件中的任何块，包括“DMNU”块、“DXDT”块，以及与音频、视频和/或字幕轨关联的用于多媒体表达的块。例如，第一多媒体文件可包括“DMNU”块(在下文详细讨论)，其引用位于第一多媒体文件的“movi”列表块之内的第一多媒体表达，以及第二多媒体文件的“movi”列表块之内的第二多媒体表达。或者，这两个的“movi”列表块能够包含在同一多媒体文件中，而“DMNU”块所在的文件不需要是同一个。

## **2.4 “DXDT”块**

【0159】“DXDT”块 38 包含了所谓的“元数据”。术语“元数据”用来描述可提供有关文件、文档或广播内容的信息。存储在根据本发明实施例的多媒体文件的“DXDT”块中的“元数据”可被用来存储内容特定信息，例如标题、作者、版权持有者以及角色。另外，将提供关于用来编码多媒体文件的编解码器的技术细节，例如使用的 CLI 选项和在每次传输后的量化因子或量化参数分配。

【0160】在一个实施例中，元数据在“DXDT”块中表现为一系列的语句，在其中每个语句包括主题、谓词、对象和权限。主题是指描述的是什么。主题可涉及文件、项目、人员或组织。主题可涉及任何具有可描述特征的事物。谓词可标识所描述的主题的特征。对象是对主题的被标识的特征的描述，以及权限标识了信息源。

【0161】下面是一个表格，其显示了“元数据”的各个部分如何被表达为主题、谓词、对象和权限的例子。

表 1 “元数据”的概念表现

主题	谓词	对象	权限
_.file281	http://purl.org/dc/elements/1.1/title	The Matrix	_.auth42
_.file281	http://xmlns.divxnetworks.com/2004/11/cast#Person	_.cast871	_.auth42
_.file281	http://xmlns.divxnetworks.com/2004/11/cast#Person	_.cast872	_.auth42
_.file281	http://xmlns.divxnetworks.com/2004/11/cast#Person	_.cast873	_.auth42
_.cast871	http://xmlns.divxnetworks.com/2004/11/cast#name	Keanu Reeves	_.auth42
_.cast871	http://xmlns.divxnetworks.com/2004/11/cast#role	Actor	_.auth42
_.cast871	http://xmlns.divxnetworks.com/2004/11/cast#character	Neo	_.auth42
_.cast282	http://xmlns.divxnetworks.com/2004/11/cast#name	Andy Wachowski	_.auth42
_.cast282	http://xmlns.divxnetworks.com/2004/11/cast#role	Director	_.auth42
_.cast283	http://xmlns.divxnetworks.com/2004/11/cast#name	Larry Wachowski	_.auth42
_.cast283	http://xmlns.divxnetworks.com/2004/11/cast#role	Director	_.auth42
_.file281	http://purl.org/dc/elements/1.1/right	Copyright 1998 warner Brothers All rights Reserved	_.auth42
_.file281	Series	_.file321	_.auth42
_.file321	Episode	2	_.auth42

_:file321	<a href="http://purl.org/dc/elements/1.1/title">http://purl.org/dc/elements/1.1/title</a>	The Matrix Reloaded	_:auth42
_:file321	Series	_:file 122	_:auth42
_:file122	Episode	3	_:auth42
_:auth42	<a href="http://xmlns.com/foaf/0.1/Organization">http://xmlns.com/foaf/0.1/Organization</a>	_:foaf92	_:auth42
_:foaf92	<a href="http://xmlns.com/foaf/0.1/name">http://xmlns.com/foaf/0.1/name</a>	Warner Brothers	_:auth42
_:file281	<a href="http://xmlns.divxnetworks.com/2004/11/track#track">http://xmlns.divxnetworks.com/2004/11/track#track</a>	_:track#dc00	_:auth42
_:track#dc00	<a href="http://xmlns.divxnetworks.com/2004/11/track#resolution">http://xmlns.divxnetworks.com/2004/11/track#resolution</a>	1024 × 768	_:auth42
_:file281	<a href="http://xmlns.divxnetworks.com/2004/11/content#certificationLevel">http://xmlns.divxnetworks.com/2004/11/content#certificationLevel</a>	HT	_:auth42
_:track#dc00	<a href="http://xmlns.divxnetworks.com/2004/11/track#frameTypeDist">http://xmlns.divxnetworks.com/2004/11/track#frameTypeDist</a>	32,1,3,5	_:auth42
_:track#dc00	<a href="http://xmlns.divxnetworks.com/2004/11/track#codecSettings">http://xmlns.divxnetworks.com/2004/11/track#codecSettings</a>	bv1 276 -psy 0 -key 300 -b 1 -sc 50 -pq 5 -vbv 6951200,3145728,235929 6 -profile 3 -nf	_:auth42

【0162】在一个实施例中，使用二进制表现的数据来实现主题、谓词、对象和权限的表达，所述二进制表现的数据可被认为构成了带标记的有向图（DLG）。DLG 包含了节点，其是资源或文字。资源是标识符，其可与命名惯例相一致，命名惯例例如由 Internet Engineering Task Force 在 RFC 2396 中定义的通用资源标识符（“URI”）（<http://www.ietf.org/rfc/rfc2396.txt>），或者是指特定于系统自身的数据。文字表现为实际数值，而不是引用。

【0163】DLG 的优点是其允许包含灵活数目的同类数据项，例如电影的角色成员。在示于表 1 的例子中，包含了三个角色成员。然而，可包括角色成员的任何数目。DLG 还允许到其它数据类型的关联的连接。在表 1 中，“元数据”项目具有主题“\_:file281”、谓词“Series”、以及对象“\_:file321”。主题“\_:file281”指示出“元数据”是指被称为“\_:file321”的文件的内容（在这个情况下，是电影“The Matrix”）。谓词是“Series”，其指示了对象将具有关于另一电影的信息，该另一电影与“The Matrix”

属于同一系列。但是，“\_:file321”不是标题或任何其他关于包含“The Matrix”的系列的特定信息，而是对提供了更多的关于“\_:file321”的信息的另一个条目的引用。然而，下一个主题为“\_:file321”的“元数据”条目包括有关“\_:file321”的数据，即由都柏林核心词汇所指定的、由“<http://purl.org/dc/elements/1.1/title>”所指示的这一续篇的标题是“The Matrix Reloaded”。

**【0164】**在表 1 中的附加的“元数据”语句指定 Keanu Reeves 是扮演角色 Neo 的成员，而 Larry 和 Andy Wachowski 是导演。技术信息也表达在“元数据”中。该“元数据”语句指定了“\_:file281”包括轨“\_:track#dc00”。“元数据”提供了包括视频轨的分辨率、视频轨的认证等级，以及编解码器设置的信息。尽管没有示于表 1 中，“元数据”还可包括在编码时指派给轨的唯一标识符。当使用唯一标识符时，对同一内容多次编码将会导致不同的标识符，其用于该内容的每个编码版本。然而，对编码视频轨的复制将保留被复制的轨的标识符。

**【0165】**示于表 1 的条目可以由其它的词汇来替代，诸如 UPnP 词汇，其由 UPnP Forum（见 <http://www.upnpforum.org>）定义。另一个替换是数字项目声明语言（DIDL）或 DIDL-Lite 词汇，其由国际标准化组织作为朝向 MPEG-21 标准的工作的一部分来发展。下文是在 UPnP 词汇之内的谓词的例子：

```
urn:schemas-upnp-org:metadata-1-0/upnp/actor
urn:schemas-upnp-org:metadata-1-0/upnp/author
urn:schemas-upnp-org:metadata-1-0/upnp/producer
urn:schemas-upnp-org:metadata-1-0/upnp/director
urn:schemas-upnp-org:metadata-1-0/upnp/genre
urn:schemas-upnp-org:metadata-1-0/upnp/album
urn:schemas-upnp-org:metadata-1-0/upnp/playlist
urn:schemas-upnp-org:metadata-1-0/upnp/originalTrackNumber
urn:schemas-upnp-org:metadata-1-0/upnp/userAnnotation
```

**【0166】**所有“元数据”的权限是“\_:auth42”。“元数据”语句显示了“\_:auth42”是“Warner Brothers”。权限使得可对文件和与文件关联的“元数据”语句的质量作出估计。

**【0167】** 图形中的节点通过命名的资源节点连接。“元数据”的语句包含了主题节点、谓词节点和对象节点。可选地，权限节点可作为“元数据”语句的一部分连接到 DLG。

**【0168】** 对于每个节点，其具有某些特征以帮助进一步地解释节点功能。使用 ANSI C 编程语言来表现可能的类型：

```

/** Invalid Type */
#define RDF_IDENTIFIER_TYPE_UNKNOWN                                0x00

/** Resource URI rdf:about */
#define RDF_IDENTIFIER_TYPE_RESOURCE                            0x01

/** rdf:NodeId, _:file or generated N-Triples */
#define RDF_IDENTIFIER_TYPE_ANONYMOUS                          0x02

/** Predicate URI */
#define RDF_IDENTIFIER_TYPE_PREDICATE                            0x03

/** rdf:li, rdf:<n> */
#define RDF_IDENTIFIER_TYPE_ORDINAL                            0x04

/** Authority URI */
#define RDF_IDENTIFIER_TYPE_AUTHORITY                            0x05

/** UTF-8 formatted literal */
#define RDF_IDENTIFIER_TYPE_LITERAL                            0x06

/** Literal Integer */
#define RDF_IDENTIFIER_TYPE_INT                                0x07

/** Literal XML data */
#define RDF_IDENTIFIER_TYPE_XML_LITERAL                        0x08

```

**【0169】** 数据结构的一个例子（以 ANSI C 编程语言表现），其在下文中表现出包含在“DXDT”块之内的“元数据”块：



```

typedef struct RDFDataStruct
{
    RDFHeader      Header;
    uint32_t       numOfStatements;
    RDFStatement   statements[RDF_MAX_STATEMENTS];
} RDFData

```

【0170】所述“RDFData”块包括一个被称为“RDFHeader”块，一个数值“numOfStatements”，以及一系列“RDFStatement”块。

【0171】所述“RDFHeader”块包括信息，其有关于“元数据”在块中被安排的形式。在一个实施例中，在“RDFHeader”块中的数据可表现如下（以ANSI C表现）：

```

typedef struct RDFHeaderStruct
{
    uint16_t       versionMajor;
    uint16_t       versionMinor;
    uint16_t       versionFix;
    uint16_t       numOfSchemas;
    RDFSchema      schemas[RDF_MAX_SCHEMAS];
} RDFHeader;

```

【0172】“RDFHeader”块包括许多“version”，其指示了资源描述格式的版本，以使得具有前向兼容性。该头包括第二数目“numOfSchemas”，其在列表“schemas”中表现了“RDFSchema”块的数目，所述列表“schemas”也组成了“RDFHeader”块的一部分。在若干个实施例中，“RDFSchema”块被用来使复杂的资源能够被更有效地表现。在一个实施例中，在“RDFSchema”块中包含的数据可表现如下（以ANSI C表现）：

```

typedef struct RDFSchemaStruct
{
    wchar_t*       prefix;
    wchar_t*       uri;
} RDFSchema;

```

**【0173】**“RDFSchema”块包括第一串文本，例如标识为“prefix”的“dc”，和第二串文本，例如标识为“uri”的“http://purl.org/dc/elements/1.1/”。所述“prefix”定义了一个可用在“元数据”中“uri”位置的项。“uri”是通用资源标识符，其与特定的标准化词汇相一致，或者是一具体系统的特定词汇。

**【0174】**回到对“RDFData”块的讨论。除了“RDFHeader”块以外，“RDFData”块还包括数值“numOfStatements”和一系列“statement”“RDFStatement”块。数值“numOfStatements”指示包含信息的“RDFStatement”块在列表“statements”中的实际数目。在一个实施例中，包含在“RDFStatement”块中的数据可表现如下（以ANSI C表现）：

```
typedef struct RDFStatementStruct
{
    RDFSubject  subject;
    RDFPredicate predicate;
    RDFObject   object;
    RDFAuthority authority;
} RDFStatement;
```

**【0175】**每个“RDFStatement”块包含一段关于多媒体文件的“元数据”。块“subject”、“predicate”、“object”和“authority”被用来包含上面所描述的“元数据”的各种各样的成分。

**【0176】**所述“subject”是“RDFSubject”块，其表现了上面描述的“元数据”的主题部分。在一个实施例中，包含在“RDFSubject”块之内的数据可表现如下（以ANSI C表现）：

```
typedef struct RDFSubjectStruct
{
    uint16_t    type;
    wchar_t*    value;
} RDFSubject;
```

【0177】上文显示的“RDFSubject”块包括一个数值“type”，其指示数据是一段“元数据”的资源还是一个匿名节点，还包括统一编码文本串“value”，其包含表现一段“元数据”的主题的数据。在已经定义了“RDFSchema”块的实施例中，所述数值可以是一定义了项，以代替直接资源引用。

【0178】在“RDFStatement”块中的“predicate”是“RDFPredicate”块，其表现了一段“元数据”的谓词部分。在一个实施例中，包含在“RDFPredicate”块之内的数据可表现如下（以ANSI C表现）：

```
typedef struct RDFPredicateStruct
{
    uint16_t    type;
    wchar_t*    value;

} RDFPredicate;
```

【0179】上文显示的“RDFPredicate”块包括了数值“type”，其指示了数据是一段“元数据”的谓词 URI 或是顺序列表条目，以及上文显示的“RDFPredicate”块包括了文本串“value”，其包含了表现为一段“元数据”的谓词的数据。在已经定义了“RDFSchema”块的实施例中，所述数值可以是一定义了项，以代替直接资源引用。

【0180】在“RDFStatement”块中的“object”是“RDFObject”块，其表现了一段“元数据”的对象部分。在一个实施例中，包含在“RDFObject”块之中的数据可表现如下（以ANSI C表现）：

```
typedef struct RDFObjectStruct
{
    uint16_t    type;
    wchar_t*    language;
    wchar_t*    dataTypeURI;
    wchar_t*    value;

} RDFObject;
```

【0181】上文显示的“RDFObject”块包括了数值“type”，其指示了该块数据是一段“元数据”的 UTF-8 文字串，整数文字，或是文字 XML 数据。该块还包括三个值。第一个数值“language”被用于表现表达该段“元数据”的语言（例如，电影的标题以不同语言变化）。在若干个实施例中，可使用标准表现来标识语言（如 RFC 3066—由 Internet Engineering Task Force 指定的语种识别标签（Tags for Identification of Language），见 <http://www.ietf.org/rfc/rfc3066.txt>）。第二个数值“dataTypeURI”被用来在如果“type”区域没有能够明确地指示时，指示包含在“value”区域之内的数据的类型。由 dataTypeURI 所指定的 URI 指向一般 RDF URI 词汇，其用来描述所使用的数据的特定类型。URI 所能被表示的不同格式在 <http://www.w3.org/TR/rdf-concepts/#section-Datatypes> 有所描述。在一个实施例中，“value”可以是“wide character（宽字符）”。在其它实施例中，“value”可能是任何从单一比特到图像或视频序列的多种数据类型。“value”包含“元数据”的对象块。

【0182】在“RDFStatement”块中的“authority”是“RDFAuthority”块，其表现了一段“元数据”的权限部分。在一个实施例中，包含在“RDFAuthority”块之内的数据可表现如下（以 ANSI C 表现）：

```
typedef struct RDFAuthorityStruct
{
    uint16_t    type;
    wchar_t*   value;
} RDFAuthority;
```

【0183】上文显示的“RDFAuthority”数据结构包括了数值“type”，其指示了数据是一段“元数据”的资源或匿名节点。“value”包含了表现“元数据”权限的数据。在已经定义了“RDFSchema”块的实施例中，所述数值可以是一定义了项，以代替直接资源引用。

【0184】图 2.3 显示了根据本发明实施例的多媒体文件的“DXDT”块的存储的概念图。该“DXDT”块 38 包括“RDFHeader”块 110，“numOfStatements”数值 112 和一系列 RDFStatement 块 114。

“RDFHeader”块 110 包括“version”数值 116、“numOfSchemas”数值 118，以及一系列“Schema”块 120。每个“RDFStatement”块 114 包括“RDFSubject”块 122、“RDFPredicate”块 124、“RDFObject”块 126，以及“RDFAuthority”块 128。“RDFSubject”块包括“type”数值 130 和“value”数值 132。“RDFPredicate”块 124 也包括“type”数值 134 和“value”数值 136。“RDFObject”块 126 包括“type”数值 138、“language”数值 140（在图中示为“lang”）、“dataTypeURI”数值 142（在图中示为“dataT”）以及“value”数值 144。“RDFAuthority”块 128 包括“type”数值 146 和“value”数值 148。尽管所描述的“DXDT”块被显示为包含单一“Schema”块和单一“RDFStatement”块，但是本领域普通技术人员可以容易地认识到在描述“元数据”的块中可以使用不同数目的“Schema”块和“RDFStatement”块。

**【0185】**如上所述，根据本发明实施例的多媒体文件可连续地修改和更新。提前确定“元数据”与文件自身关联以及“元数据”远程访问（例如，通过互连网）是困难的。通常，足够的“元数据”被包含在根据本发明实施例的多媒体文件中，以描述文件的内容。如果检查该文件的装置可通过网络访问其它装置，则可获得额外信息，其中所述其它装置包含从该文件内引用的“元数据”。

**【0186】**上述表现“元数据”的方法可以是可扩展的，并可提供这样的能力：其可根据它随时间改变的需要，增加和去除存储在文件之内的不同“元数据”区域。此外，“元数据”的表现形式可在修订版本间前向兼容。

**【0187】**“元数据”根据本发明实施例所表现出的结构化形式使得查询多媒体文件的装置可以更好地确定多媒体文件的内容。该查询可接下来用于更新多媒体文件的内容，以获得额外的关于多媒体文件的“元数据”，产生涉及文件内容的菜单或执行任何其它功能，这些功能包含在以标准格式表现的数据的自动化处理中。另外，限定“元数据”的每个可分解的分量的长度，可增加具有有限数量内存的装置可访问“元数据”的轻易性，所述具有有限数量内存的装置例如消费类电子装置。

【0188】在其它实施例中，使用每段“元数据”的个体块来表现“元数据”。若干个根据本发明的“DXDT”包括包含了如上文所描述被编码的“元数据”的二进制块，以及包含了按上文所描述的格式或其它格式而被格式化的个体“元数据”段的额外块。在二进制“元数据”被包含在“DXDT”块的实施例中，可使用64位编码的ASCII来表现所述二进制“元数据”。在其它实施例中，可使用其它的二进制表现形式。

【0189】在图2.3.1中示出了可被包含在根据本发明的“DXDT”块中的个体块的例子。该“元数据”包括“MetaData”块150，其可包括“PixelAspectRatioMetaData”块152a、“EncoderURIMetaData”块152b、“CodecSettingMetaData”块152c、“FrameTypeMetaData”块152d、“VideoResolutionMetaData”块152e、“PublisherMetaData”块152f、“CreatorMetaData”块152g、“GenreMetaData”块152h、“CreatorToolMetaData”块152i、“RightsMetaData”块152j、“RunTimeMetaData”块152k、“QuantizerMetaData”块152l、“CodecInfoMetaData”块152m、“EncoderNameMetaData”块152n、“FrameRateMetaData”块152o、“InputSourceMetaData”块152p、“FileIDMetaData”块152q、“TypeMetaData”块152r、“TitleMetaData”块152s和/或“CertLevelMetaData”块152t。

【0190】“PixelAspectRatioMetaData”块152a包括关于编码的视频像素长宽比的信息。“EncoderURIMetaData”块152b包括关于编码器的信息。“CodecSettingMetaData”块152c包括关于用于编码视频的编解码器的设置的信息。“FrameTypeMetaData”块152d包括关于视频帧的信息。“VideoResolutionMetaData”块152e包括关于编码视频的视频分辨率的信息。“PublisherMetaData”块152f包括关于发布媒体的人员或组织的信息。“CreatorMetaData”块152g包括关于内容创建者的信息。“GenreMetaData”块152h包括关于媒体流派（genre）的信息。“CreatorToolMetaData”块152i包括关于用于产生文件的工具的信息。“RightsMetaData”块152j包括关于DRM的信息。“RunTimeMetaData”块152k包括关于媒体运行时间的信息。“QuantizerMetaData”块152l

包括关于用于编码视频的量化因子的信息。“CodecInfoMetaData”块 152m 包括关于编解码器的信息。“EncoderNameMetaData”块 152n 包括关于编码器的名称的信息。“FrameRateMetaData”块 152o 包括关于媒体的帧率的信息。“InputSourceMetaData”块 152p 包括关于输入源的信息。“FileIDMetaData”块 152q 包括文件的唯一标识符。“TypeData”块 152r 包括关于多媒体文件类型的信息。“TitleMetaData”块 152s 包括了媒体的标题，以及“CertLevelMetaData”块 152t 包括关于媒体认证等级的信息。在其它实施例中，能够包含额外的块，其包含额外的“元数据”。在若干个实施例中，上文所描述的二进制格式包含“元数据”的块可被包含在“MetaData”块之内。在一个实施例中，二进制“元数据”可以 64 位 ASCII 来编码。

## 2.5 “DMNU” 块

**【0191】**参考图 2.0 和 2.0.1，显示了第一“DMNU”块 40 (40') 和第二“DMNU”块 46 (46')。在图 2.0 中，第二“DMNU”块 46 组成了多媒体文件 30 的一部分。在图 2.0.1 中所示的实施例中，“DMNU”块 46' 包含在分离的 RIFF 块之内。在两个例子中，第一和第二“DMNU”块包含了可被用来显示导航菜单的数据。在一个实施例中，第一“DMNU”块 40 (40') 包含了可用于创建简单菜单的数据，该简单菜单不含有诸如扩展背景动画的高级功能部件。另外，第二“DMNU”块 46 (46') 包含了可用于创建更复杂菜单的数据，该更复杂的菜单包括诸如扩展动画背景的高级功能部件。

**【0192】**在若干个实施例中，提供简单和复杂菜单可使得装置能够选择所希望显示的菜单。将两个菜单中较小的一个放置在“movi”列表块 42 之前，使得根据本发明实施例的、不显示菜单的装置能够快速略过不能被显示的信息。

**【0193】**在其它实施例中，创建简单菜单所需的数据在第一和第二“DMNU”块之间分开。或者，“DMNU”块可以是“movi”块之前的、包含单组菜单或多组菜单数据的单一块。在其它实施例中，“DMNU”块可以是位于贯穿多媒体文件的其它位置的单一块或多个块。

【0194】在几个根据本发明的多媒体文件中，第一“DMNU”块 40(40')可基于第二“DMNU”块 46(46')中的“richer”菜单自动创建。下文将更详细地讨论菜单的自动创建。

【0195】图 2.4 中显示了根据本发明实施例的“DMNU”块的结构。该“DMNU”块 158 是包含菜单块 160 和“MRIF”块 162 的列表块。菜单块包含了信息，其对于构造菜单和通过菜单导航是必要的。“MRIF”块包括了媒体信息，其能够用来提供菜单的字幕、背景视频和背景音频。在多个实施例中，“DMNU”块包含了菜单信息，其使得菜单能够以几种不同的语言来显示。

【0196】在一个实施例中，“WowMenu”块 160 包括了菜单块对象的层次结构，其在图 2.5 中概念性地被示出。在该层次结构的顶端是“WowMenuManager”块 170。该“WowMenuManager”块能够包含一个或多个“LanguageMenus”块 172 和一个“Media”块 174。

【0197】“LanguageMenus”块 172 的使用使得“DMNU”块 158 可以不同语言包含菜单信息。每个“LanguageMenus”块 172 包含用来以某种具体语言来产生一组完整菜单的信息。因此，“LanguageMenus”块包含了标识符，其标识与“LanguageMenus”块关联的信息的语言。“LanguageMenus”块还包括一系列“WowMenu”块 175。

【0198】每个“WowMenu”块 175 包括了全部的信息，这些信息将被显示在屏幕上，用于特定菜单。这些信息还可包括背景视频和音频。所述信息还可包括关于按钮动作的数据，其能够用来访问其它菜单或退出菜单并开始播放多媒体文件的一部分。在一个实施例中，“WowMenu”块 175 包括一系列媒体引用。这些引用涉及到包含在“Media”块 174 中的信息，其将在下文进一步描述。媒体引用可定义菜单的背景视频和音频。“WowMenu”块 175 还定义了一覆盖物，其可被用来在菜单被第一次访问时，高亮特定的按钮。

【0199】另外，每个“WowMenu”块 175 包括许多“ButtonMenu”块 176。每个“ButtonMenu”块 176 定义了显示器上的按钮的性质。“ButtonMenu”块可描述这样的情况：当用户高亮按钮时使用的覆盖物、按钮的名字，以及如何响应用户通过菜单导航来执行的不同动作。



通过引用“Action”块 178 来定义对动作的响应。一个单一的动作，例如，选定按钮，可以引起几个“Action”块被访问。在用户能够通过使用诸如鼠标之类的、使显示器上的指示器或指针以不受约束的形式围绕显示器移动的装置而与菜单交互的实施例中，按钮在显示器上的位置可使用“MenuRectangle”块 180 来定义。对按钮在显示器上位置的认识，使得系统能够在使用自由范围的输入装置时，确定用户是否选择了按钮。

**【0200】**每个“Action”块标识了动作相关块的许多不同变化中的一个或多个，该动作相关块可包括“PlayAction”块 182、“MenuTransitionAction”块 184、“ReturnToPlayAction”块 186、“AudioSelectAction”块 188、“SubtitleSelectAction”块 190，以及“ButtonTransitionAction”块 191。“PlayAction”块 182 标识了多媒体文件之内的每个视频轨、音频轨和字幕轨的一部分。该“PlayAction”块通过对“MediaTrack”块的引用（见下文的讨论）来引用视频轨的一部分。“PlayAction”块使用“SubtitleTrack”192 和“AudioTrack”块 194 标识了音频和字幕轨。该“SubtitleTrack”和“AudioTrack”块都包含了“MediaTrack”198 的引用。当“PlayAction”块根据本发明实施例构成了动作的基础时，所选择的音频和字幕轨由变量集的值来确定，该变量集初始被设置为默认值，并接着通过用户和菜单的交互而潜在地被改变。

**【0201】**每个“MenuTransitionAction”块 184 包含了“WowMenu”块 175 的引用。该引用能够用来获得转换到并显示另一菜单的信息。

**【0202】**“ReturnToPlayAction”块 186 包含信息，其使得播放器可返回到多媒体文件的一部分，该部分在用户提出菜单前正被访问。

**【0203】**“AudioSelectAction”块 188 包含了信息，其可被用来选择一特定的音频轨。在一个实施例中，音频轨从包含在根据本发明实施例的多媒体文件之内的音频轨中被选择。在其它实施例中，音频轨可位于外部引用文件中。

**【0204】**“SubtitleSelectAction”块 190 包含信息，其可被用来选择特定的字幕轨。在一个实施例中，字幕轨选自包含在根据本发明实施例

的多媒体文件之内的字幕。在其它实施例中，字幕轨可位于外部引用文件中。

**【0205】** 每个“ButtonTransitionAction”块 191 包含信息，其可被用来转换到同一菜单中的另一个按钮。这将在其它与按钮关联的动作已被执行之后才执行。

**【0206】** “Media”块 174 包括许多“MediaSource”块 166 和“MediaTrack”块 198。“Media”块定义了所有由功能部件（feature）和菜单系统所使用的多媒体轨（例如，音频、视频、字幕）。每个“MediaSource”块 196 标识了根据本发明实施例的多媒体文件之内的 RIFF 块，其还可包括多个 RIFF 块。每个“MediaTrack”块 198 标识了 RIFF 块之内的多媒体轨的一部分，该 RIFF 块由“MediaSource”块来指定。

**【0207】** “MRIF”块 162 本质上是它自己的小多媒体文件，其与 RIFF 相兼容。该“MRIF”块包括了音频、视频和字幕轨，其可被用来为菜单提供背景音频和视频以及覆盖物。该“MRIF”块还能够包含用作覆盖物的视频，以指示被高亮的菜单按钮。在需要较少菜单数据的实施例中，背景视频可以是静止帧（AVI 格式的变形）或小序列标识帧。在其它实施例中，更精细的视频序列可被用来提供背景视频。

**【0208】** 如上所述，组成了“WowMenu”块 175 的部分的各种块以及“WowMenu”块自身包含当前媒体轨的引用。每个这些引用通常是针对于定义在 RIFF 块的“hdrl”列表块中的多媒体轨的。

**【0209】** 在图 2.6 中显示了其它可被用于创建根据本发明的“DMNU”块的块。“DMNU”块包含了“WowMenuManager”块 170'。该“WowMenuManager”块 170'能够包含至少一个“LanguageMenus”块 172'，至少一个“Media”块 174'，以及至少一个“TranslationTable”块 200。

**【0210】** “LanguageMenus”块 172'的内容非常类似于图 2.5 示出的“LanguageMenus”块 172。主要的不同在于“PlayAction”块 182'没有包含“SubtitleTrack”块 192 和“AudioTrack”块 194。

【0211】“Media”块 174'与示于图 2.5 的“Media”块 174 有很大不同。“Media”块 174'包含至少一个“Title（标题）”块 202 和至少一个“MenuTracks”块 204。“Title”块是指多媒体文件之内的标题。如上所述，根据本发明实施例的多媒体文件包括多于一个的标题（例如，电视连续剧的多个片断，一系列相关的全长影片（feature）或仅仅不同影片的选择）。“MenuTracks”块 204 包含了关于媒体信息的信息，该媒体信息用来创造菜单显示以及伴随着显示的音频轨和字幕。

【0212】“Title”块能够包含至少一个“Chapter（章节）”块 206。“Chapter”块 206 是指特定标题之内的场景。“Chapter”块 206 包括视频轨、每个音频轨和每个字幕轨的一部分的引用，所述视频轨、每个音频轨和每个字幕轨相应于由“Chapter”块所指示的场景。在一个实施例中，引用可使用“MediaSource”块 196'和“MediaTrack”块 198'来实现，所述“MediaSource”块 196'和“MediaTrack”块 198'类似于那些上文相应于图 2.5 所描述那些。在若干个实施例中，“MediaTrack”块引用视频轨的合适部分，而许多额外的“MediaTrack”块每个都引用音频轨或字幕轨中的一个。在一个实施例中，所有对应于特定视频轨的音频轨和字幕轨使用分离的“MediaTrack”轨来引用。

【0213】如上所述，“MenuTrack”块 204 包含媒体的引用，其用来产生菜单的音频、视频和覆盖媒体。在一个实施例中，媒体信息的引用通过使用包含在“MenuTrack”块之内的“MediaSource”块 196'和“MediaTrack”块 198'来获得。在一个实施例中，“MediaSource”块 196'和“MediaTrack”块 198'以上文描述的与图 2.5 关联的方式实现。

【0214】“TranslationTable”块 200 可被用来容纳文本串，其以多种语言描述了每个标题和章节。在一个实施例中，“TranslationTable”块 200 包括至少一个“TranslationLookup”块 208。每个“TranslationLookup”块 208 与“Title”块 202、“Chapter”块 206 或“MediaTrack”块 196'关联，并包含了许多个“Translation”块 210。“TranslationLookup”块中的每个“Translation”块包含了文本串，其以“Translation”块所指示的语言来描述与“TranslationLookup”块关联的块。

【0215】图 2.6.1 中示出了是概念图，其示出了包含在“DMNU”块之内的不同的块之间的关联。该图中使用实箭头来显示一个块被另一个块所包含。箭头所指的方向指示了该块包含于该箭头所发源的块。通过虚线来指示从一个块到另一个块的引用，其中所被引用的块由虚线箭头所指示。

## 2.6 “junk” 块

【0216】“junk”块 41 是一个可选的块，其可被包含在根据本发明实施例的多媒体文件中。在 AVI 文件格式中指定了“junk”块的特性。

## 2.7 “movi” 列表块

【0217】“movi”列表块 42 包含了许多“数据”块。“数据”块能够包含的信息的例子有音频、视频或字幕数据。在一个实施例中，“movi”列表块包括至少一个视频轨数据、多个音频轨和多个字幕轨的数据。

【0218】图 2.7 示出了包含一个视频轨、三个音频轨和三个字幕轨的多媒体文件的“movi”列表块 42 中的“数据”块的交错。为了便于描述，包含视频的“数据”块将被描述为“视频”块，包含音频的“数据”块将称为“音频”块，以及包含字幕的“数据”块将被称为“字幕”块。在所描述的“movi”列表块 42 中，每个“视频”块 262 通过来自每个音频轨的“音频”块 264，而与下一个“视频”块相分离。在若干个实施例中，“音频”块包含了音频轨的部分，该部分相应于跟随在该“音频”块之后的“视频”块中所包含的视频部分。

【0219】相邻的“视频”块也可通过一个或多个来自字幕轨之一的“字幕”块 266 而分离。在一个实施例中，“字幕”块 266 包含字幕以及开始时间和结束时间。在若干个实施例中，“字幕”块这样交错在“movi”列表块中：跟随着“字幕”块的“视频”块包含视频部分，该视频部分发生在该字幕的开始时间。在其它实施例中，所有“字幕”和“音频”块的开始时间领先于相应视频的开始时间。在一个实施例中，“音频”和“字幕”块可被放置在相应的“视频”块的 5 秒之内，而在其它实施例中，“音频”和“字幕”块可被放置在一定时间内，该时间与

视频能够通过可以显示文件之内的音频和视频的装置，而被缓冲的数量有关。

【0220】在一个实施例中，“数据”块包括“FOURCC”码，以标识“数据”块所属的流。“FOURCC”码由跟随有双字符码的两数字流数（two-digit stream number）构成，所述双字符码定义了块中的信息的类型。（侵入者）一个可替换的“FOURCC”码由跟随有两数字流数的双字符码构成，其中该双字符码定义了块中信息的类型。在下表中显示了双字符码的例子：

表 2 使用在 FOURCC 编解码器中的选定的双字符码

双字符码	描 述
db	未压缩的视频帧
dc	压缩的视频帧
dd	视频帧的 DRM 密钥信息
pc	调色板变化
wb	音频数据
st	字幕（文本模式）
sb	字幕（位图模式）
ch	章节

【0221】在一个实施例中，“视频”块 262 和“音频”块 264 的结构与 AVI 文件格式兼容。在其它实施例中，可使用块的其它格式，其指定了媒体的特性并包含着编码的媒体。

【0222】在若干个实施例中，包含在“字幕”块 266 之内的数据可表现如下：

```
typedef struct _subtitlechunk {
    FOURCC fcc;
    DWORD cb;
    STR duration;
    STR subtitle;
} SUBTITLECHUNK;
```

【0223】数值“fcc”是 FOURCC 码，其表示字幕轨和字幕轨的特性（文本或位图模式）。数值“cb”指定了结构的尺寸。数值“duration”指定了字幕的开始和结束点的时间。在一个实施例中，数值“duration”可以是 hh:mm:ss.xxx-hh:mm:ss.xxx 的形式。hh 代表小时，mm 代表分钟，ss 是秒，以及 xxx 是毫秒。数值“subtitle”包含了文本模式下的字幕的统一编码文本，或者包含了位图模式下的字幕的位图图像。本发明的几个实施例使用了压缩位图图像来表现字幕信息。在一个实施例中，“字幕”区域包含了信息，其是关于字幕的宽度、高度和显示器上的位置的。另外，“字幕”区域还能够包含位图的颜色信息和现有像素。在若干个实施例中，使用游程编码或游程长度编码来减少表现位图所需的像素信息量。

【0224】根据本发明实施例的多媒体文件可包括数字版权管理。这一信息可用在视频点播应用中。由数字版权管理保护的多媒体文件仅仅在获得了特定回放版权许可的播放器上可被正确地回放。在一个实施例中，轨受到数字版权管理保护的事实可以在信息中被表示，该信息是有关于磁轨的，并处于“hdl”列表块中（见上文的描述）。包含受到数字版权管理保护的磁轨、根据本发明实施例的多媒体文件还可在“movi”列表块中包含有关数字版权管理的信息。

【0225】在图 2.8 中示出了根据本发明实施例的多媒体文件的“movi”列表块，其包括一个视频轨、多个音频轨、至少一个字幕轨，以及使能数字版权管理的信息。该“movi”列表块 42' 类似于显示在图 2.7 中的“movi”列表块，但在每个视频块 262' 之前增加了“DRM”块 270，“DRM”块 270 是“数据”块，其包含了数字版权管理信息，其可通过 FOURCC 码“nndd”来标识。前两个字符“nn”代表轨数，以及最后两个字符是“dd”，其意味着块包含了数字版权管理信息。在一个实施例中，“DRM”块 270 提供了用于跟随着“DRM”块的“视频”块 262' 的数字版权管理信息。试图播放数字版权管理保护的视频轨的装置使用了“DRM”块中的信息，来解码“视频”块中的视频信息。通常，在“视频”块之前没有“DRM”块被解释为“视频”块没有受到保护的含义。

【0226】在根据本发明实施例的加密系统中，视频块仅仅部分加密。当使用部分加密时，“DRM”块包含了被加密的“视频”一部分的引用，以及可用于解密加密部分的密钥的引用。解密密钥可位于“DRM”头中，其是“strd”块（见上文描述）的一部分。该解密密钥通过主密钥来编码和加密。“DRM”头还包含了标识主密钥的信息。

【0227】图 2.9 显示了“DRM”块中信息的概念表现。“DRM”块 270 包括了“frame”数值 280、“status”数值 282、“offset”数值 284、“number”数值 286，以及“key”数值 288。“frame”数值能够用来代表加密视频帧。“status”数值能够用来指示帧是否被加密，“offset”数值指向帧内加密块的起始位置，以及“number”数值指示了该块中加密的字节数目。“key”数值 288 代表能够用来解密所述块的解密密钥。

## 2.8 “idx1”块

【0228】“idx1”块 44 是可选的块，其用来在“movi”列表块 42 中索引“数据”块。在一个实施例中，“idx1”块可特定在 AVI 格式中被实现。在其它实施例中，“idx1”块可使用数据结构来实现，该数据结构代表了“movi”列表块中的每个“数据”块的文件之内的位置。在若干个实施例中，“idx1”块通过数据的轨数和数据类型，来标识每个“数据”块。前文提到的 FOURCC 码可用于此目的。

## 3. 编码多媒体文件

【0229】本发明的实施例能够用来以多种方式产生多媒体文件。在一个实例中，根据本发明实施例的系统能够从包含分离视频轨、音频轨和字幕轨的文件中产生多媒体文件。在这样的实例中，其它信息（例如菜单信息和“元数据”）可被创编并插入到文件中。

【0230】其它根据本发明实施例的系统能够用来从数字视频光盘（“DVD”）上的各种文件中提取信息，创编出根据本发明实施例的单一多媒体文件。在 DVD 为初始信息源的情况下，根据本发明实施例的系统能够使用编解码器来获得更大的压缩，并且能够对音频进行重新分块，以使音频块对应于新创建的多媒体文件中的视频块。另外，能

够对 DVD 菜单系统中的菜单信息进行语法分析，并将其用来产生多媒体文件中所包括的菜单信息。

**【0231】**其它实施例通过将额外的内容添加到根据本发明实施例的现有多媒体文件中，可产生新的多媒体文件。添加额外内容的例子可以是将额外的音频轨诸如包含解说词（例如导演评论、休假视频的后期叙述）的音频轨添加到文件中。交错在多媒体文件中的额外音频轨信息还可与多媒体文件中的菜单信息一起被修改，从而能够播放新的音频轨。

### 3.1 使用存储数据轨来产生

**【0232】**在图 3.0 中示出了根据本发明实施例的系统，其用于产生多媒体文件。系统 350 的主要组成部分是交错器 352。该交错器接收块信息，并将它们交错，从而以上文所述的格式来产生根据本发明实施例的多媒体文件。该交错器还从元数据管理器 354 接收关于“元数据”的信息。该交错器将根据本发明实施例的多媒体文件输出到存储装置 356。

**【0233】**提供到交错器的块通常被存储在存储装置上。在若干个实施例中，所有的块存储在同一个存储装置上。在其它实施例中，块可从多个存储装置提供到交错器，或者实时产生并提供到交错器。

**【0234】**在图 3.0 所示的实施例中，“DMNU”块 358 和“DXDT”块 360 已经被产生并被存储在存储装置上。视频源 362 存储在存储装置上，并使用视频解码器 364 来解码，然后使用视频编码器 366 编码，以产生“视频”块。音频源 368 也存储在存储装置上。以如下方式产生音频块：使用音频解码器 370 来解码音频源，然后使用音频编码器 372 编码该解码的音频。从存储在存储装置上的文本字幕 374 来产生“字幕”块。该字幕被提供给第一代码转换器 376，其将许多字幕格式中的任何格式转换为原始位图格式。在一个实施例中，所存储的字幕格式可以是诸如 SRT、SUB 或 SSA 的格式。另外，位图格式可以是四位位图，其包含调色板查找表。该调色板查找表包括用于 16 种可能的四位颜色编码的 24 位色深标识。一单一的多媒体文件可包括多于一个的调色板查找表（见上文表 2 中的“pc”调色板 FOURCC 码）。从而，



该四位位图允许菜单具有来自一千六百万颜色的调色板的 16 种不同的同时颜色。在替换性实施例中，使用了每像素不同的比特数和不同的色深。第一代码转换器 376 的输出被提供给第二代码转换器 378，其压缩位图。在一个实施例中，使用游程编码来压缩位图。在其它实施例中，使用了其它合适的压缩格式。

【0235】在一个实施例中，各种编码器、解码器及代码转换器之间的接口与 Microsoft Corporation 指定的 Direct Show 标准相一致。在其它实施例中，用来执行编码、解码和代码转换的软件不需要和这些标准兼容。

【0236】在描述性实施例中，分离的处理分量被显示，其用于每个媒体源。在其它实施例中，资源可被共享。例如，一单一的音频解码器和音频编码器能够用来从所有的源产生音频块。通常，整个系统可通过使用软件在计算机上实现，并且可被连接到诸如硬盘驱动器的存储装置。

【0237】为了以上文所述的方式来使用交错器，必须产生根据本发明实施例的“DMNU”块、“DXDT”块、“视频”块、“音频”块，以及“字幕”块，并将其提供给交错器。在根据本发明实施例的多媒体文件中产生各种块的每一个的处理过程，将在下文中非常详细地讨论。

### 3.2 产生“DXDT”块

【0238】“DXDT”块可通过多种方式来产生。在一个实施例中，“元数据”块通过图形用户界面进入到数据结构中，并接着语法分析到“DXDT”块。在一个实施例中，“元数据”表现为一系列的主题、谓词、对象和权限语句。在另一个实施例中，“元数据”语句可表现为多种格式中的任意格式。在若干个实施例中，每个“元数据”语句被语法分析成分离的块。在其它实施例中，几个具有第一格式（例如，主题、谓词、对象、权限表达式）的“元数据”语句被语法分析到第一块，以及其它具有其它格式的“元数据”语句被语法分析到分离的块。在一个实施例中，“元数据”语句被写入 XML 配置文件中，并且该 XML 配置文件被进行语法分析以在“DXDT”块之内创建块。

【0239】图 3.1 显示了一个系统，其用于从一系列包含在 XML 配置文件中的“元数据”语句，来产生“DXDT”块。系统 380 包括 XML 配置文件 382，其可被提供给语法分析器 384。该 XML 配置文件包括被编码为 XML 的“元数据”。该语法分析器对 XML 进行语法分析，并通过将“元数据”语句转换为块，而产生“DXDT”块 386，上述“元数据”语句所转换的块可根据上文描述的任何“元数据”块格式而被写入“DXDT”块。

### 3.3 产生“DMNU”块

【0240】图 3.2 示出了能够用来产生根据本发明实施例的“DMNU”块的系统。菜单块产生系统 420 需要输入媒体模型 422 和媒体信息。媒体信息可具有如下形式：视频源 424、音频源 426，以及覆盖源 428。

【0241】使用菜单块产生系统的输入的“DMNU”块的产生包括创建多个中间文件。媒体模型 422 用来产生 XML 配置文件 430，媒体信息用来创建多个 AVI 文件 432。XML 配置文件靠模型代码转换器 434 来创建。AVI 文件 432 通过使用交错器 436 交错视频、音频和覆盖信息，而被创建。通过使用视频解码器 438 和视频编码器 440，解码视频源 424 并以下文描述的方式再次编码该解码的视频源，从而获得视频信息。通过使用音频解码器 442 和音频编码器 444，解码音频并以下文描述的方式编码该解码的音频，从而获得音频信息。使用第一代码转换器 446 和第二代码转换器 448 来产生覆盖信息。第一代码转换器 446 将覆盖物转换为图形表达，例如标准位图，而第二代码转换器接收图形信息并根据需要对其格式化以便包含在多媒体中。当包含构建菜单所需信息的 XML 文件和 AVI 文件已经产生时，菜单产生器 450 就可使用所述信息来产生“DMNU”块 358’。

#### 3.3.1 菜单模型

【0242】在一个实施例中，媒体模型是一个面向对象的模型，其代表了所有的菜单和它们的子组成部分。媒体模型将菜单组织为一层次结构，其允许菜单靠语言选择被组织。图 3.3 描述了根据本发明实施例的媒体模型。媒体模型 460 包含顶层“MediaManager”对象（top-level

“MediaManager” object) 462, 其与许多“LanguageMenus”对象 463、“Media”对象 464, 以及“TranslationTable”对象 465 关联。“Menu Manager”还包含了默认的菜单语言。在一个实施例中, 默认语言可能由 ISO 639 两字母语言代码。

【0243】“LanguageMenus”对象依靠语言选择来组织用于不同菜单的信息。对于一种给定语言的所有的“Menu”对象 466 都与该种语言的“LanguageMenus”对象 463 关联。每个“Menu”对象与多个“Button”对象关联, 并引用了多个“MediaTrack”对象 488。所引用的“MediaTrack”对象 488 指示了“Menu”对象 466 的背景视频和背景音频。

【0244】每个“Button”对象 468 与“Action”对象 470 和“Rectangle”对象 484 关联, 该“Button”对象 468 还包含“MediaTrack”对象 488 的引用, 其指示了在按钮在显示器上高亮时所使用的覆盖物。每个“Action”对象 470 都与多个对象关联, 其可包括“MenuTransition”对象 472、“ButtonTransition”对象 474、“ReturnoPlay”对象 476、“SubtitleSelection”对象 478、“AudioSelection”对象 480, 以及“PlayAction”对象 482。每个这些对象定义了菜单系统对于来自用户的各种输入的响应。“MenuTransition”对象包含对“Menu”对象的引用, 其指示一个将响应于动作而转变的菜单。“ButtonTransition”对象指示一个将响应于动作而高亮的按钮。“ReturnoPlay”对象可引起播放器重新播放影片。“SubtitleSelection”和“AudioSelection”对象包含了“Title”对象 487 (在下文讨论) 的引用。“PlayAction”对象包含了对“Chapter”对象 492 的引用。“Rectangle”对象 484 指示了被按钮所占据的显示器的部分。

【0245】“Media”对象 464 指示了菜单系统中所引用的媒体信息。该“Media”对象具有与之关联的“MediaTracks”对象 486 和许多“Title”对象 487。“MediaTracks”对象 486 引用了“MediaTrack”对象 488, 该“MediaTrack”对象 488 是用于构建菜单的媒体 (即, 背景音频、背景视频和覆盖物) 的指示。

【0246】“Title”对象 487 是多媒体影像 (presentation) 的表示, 并具有许多与其关联的“Chapter”对象 492 和“MediaSource”对象 490。

“Title”对象还包含了对“TranslationLookup”对象 494 的引用。“Chapter”对象表现了多媒体影像上的特定点，并具有许多与之关联的“MediaTrack”对象 488。“Chapter”对象还包含引用“TranslationLookup”对象 494。每个与“Chapter”对象关联的“MediaTrack”对象指示了多媒体影像的音频、视频或字幕轨中的一点，并引用“MediaSource”对象 490 和“TranslationLookup”对象 494。

【0247】“TranslationTable”对象 465 将许多文本串组合，所述文本串描述了由“Title”对象、“Chapter”对象和“MediaTrack”对象所指示的多媒体影像的各种各样的部分。“TranslationTable”对象 465 具有许多与之关联的“TranslationLookup”对象 494。每个“TranslationLookup”对象是特定对象的指示，并具有许多与之关联的“Translation”对象 496。“Translation”对象指示每个文本串，该文本串以具体语言描述了由“TranslationLookup”对象所指示的对象。

【0248】媒体对象模型可使用软件来构建，该软件被配置，以产生各种各样上文所描述的对象，并建立对象之间所需的关联和引用。

### 3.3.2 产生XML文件

【0249】XML 配置文件产生自菜单模型，其代表了所有的菜单和它们的子分量。该 XML 配置文件还标识了所有用于菜单的媒体文件。通过实现合适的语法分析器应用程序来产生该 XML，所述语法分析器应用程序将对象模型语法分析为 XML 码。

【0250】在其它实施例中，视频编辑应用程序可为用户提供一用户界面，以使得在不创建菜单模型的情况下可以直接产生 XML 配置文件。

【0251】在另一个菜单系统是诸如 DVD 菜单的菜单模型基础的实施例中，用户能够删减菜单，以去除那些与没有包含在根据本发明实施例的多媒体文件中的内容相关联的菜单选项。在一个实施例中，这可通过提供图形用户界面来实现，该图形用户界面使得能够从菜单模型中去除对象。在另一个实施例中，可通过提供图形用户界面或可编辑 XML 配置文件的文本界面而实现删减菜单。

### 3.3.3 媒体信息

【0252】当“DMNU”块产生时，提供给菜单产生器 450 的媒体信息包括所需的数据，以提供背景视频、背景音频和前景（foreground）覆盖物，其用于菜单模型（见上文的描述）中特定的按钮。在一个实施例中，视频编辑应用程序，例如由位于 Santa Clara, CA 的 Roxio, Inc. 发布的 VideoWave，被用来提供源媒体轨，其表现为视频、音频和按钮选择覆盖物，用于每个个体的菜单。

### 3.3.4 产生中间AVI文件

【0253】如上所述，被用来作为背景视频、背景音频和前景按钮覆盖物的媒体轨被存储在单一的 AVI 文件中，用于一个或多个菜单。包含着在菜单 AVI 文件中媒体轨的块可通过使用软件来创建，该软件被设计用来交错视频、音频和按钮覆盖轨。所述“音频”、“视频”和“overlay”块（即，包含覆盖信息的“字幕”块）使用交错器而被交错到 AVI 格式兼容文件中。

【0254】如上文所提到的，分离的 AVI 文件可被创建，用于每个菜单。在其它实施例中，其它文件格式或单一文件可被用来包含媒体信息，该媒体信息用于提供背景音频、背景视频和前景覆盖信息。

### 3.3.5 将XML配置文件和AVI文件结合

【0255】在一个实施例中，计算机被配置，以对来自 XML 配置文件的信息进行语法分析，从而创建“WowMenu”块（上文描述）。另外，计算机可使用包含每个菜单媒体的 AVI 文件，来创建“MRIF”块（上文描述）。计算机接着可通过创建必要的引用，来完成“DMNU”块的创建，所述引用是在“WowMenu”块和“MRIF”块中的媒体块之间的。在若干个实施例中，菜单信息可被加密。可通过类似于下文描述的、与“视频”块有关的方式加密媒体信息，从而完成加密。在其它实施例中，可使用各种各样的可替换的加密技术。

### 3.3.6 从对象模型自动产生菜单

【0256】参考回图 3.3，与全部菜单相比，包含较少内容的菜单可通过仅仅检查与“Media”对象 464 相关联的“Title”对象 487，而自动从菜单模型产生。在图 3.3.1 中显示了被用来自动产生根据本发明实施例

的菜单的对象。软件可产生简单菜单的 XML 配置文件，该简单菜单仅仅使能对多媒体影像的具体片段的选择以及对所使用的音频和字幕轨的选择。

### 3.3.7 使用一单一配置文件来产生“DXDT”和“DMNU”块

【0257】根据本发明的几个实施例的系统具有产生包含了“元数据”和菜单信息的单一 XML 配置文件，并使用该 XML 配置文件产生“DXDT”和“DMNU”块的能力。这些系统使用“元数据”信息和菜单对象模型获得 XML 配置文件。在其它实施例中，配置文件不需要是 XML 的。

## 3.4 产生“音频”块

【0258】通过解码音频源并接着根据本发明的实践将源编码成为的“音频”块，可产生根据本发明实施例的多媒体文件的“movi”列表块中的“音频”块。在一个实施例中，所述“音频”块可使用 mp3 编解码器来被编码。

### 3.4.1 重新分块音频

【0259】当音频源被提供在块中时，其并未包含相应于相应“视频”块内容的音频信息，于是本发明可重新分块音频。图 3.4 中示出了可用于重新分块音频的处理过程。处理过程 480 包括标识（482）“视频”块、标识（484）伴随着“视频”块的音频信息，以及从现有的音频块中提取（486）音频信息，以产生（488）新的“音频”块。该处理过程一直重复，直到做出整个音频源已经被重新分块的决定（490）为止。在这个时候，音频的重新分块完成（492）。

## 3.5 产生“视频”块

【0260】如上所述，产生视频块的处理过程可包括解码视频源，以及编解码的视频到“视频”块。在一个实施例中，每个“视频”块包含用于单一视频帧的信息。解码处理过程仅仅包含以特定格式接受视频，并将视频从该格式解码为标准视频格式，其可以是未压缩的。编

码处理过程包括接受该标准视频，编码该视频，并使用编码的视频产生“视频”块。

**【0261】**图 3.5 中概念性地示出了根据本发明实施例的视频编码器。视频编码器 500 预处理 502 标准视频信息 504。接着在预处理的视频上执行运动估计 506，以向预处理视频提供运动补偿 508。在运动补偿视频上执行离散余弦变换（DCT 变换）510。在 DCT 变换后，视频被量化 512，接着执行预测 514。然后，通过将纹理编码（Texture Coded）518 版本的视频与使用运动估计结果所产生的运动编码 520 相结合，从而产生压缩的比特流 516。接着，压缩的比特流被用来产生“视频”块。

**【0262】**为了执行运动估计 506，系统必须了解预先处理的视频帧将如何通过解码装置被解码（例如，当播放者对压缩的视频进行解压缩以便观看时）。通过对量化器 512 的输出加以反量化 522，可获得这一信息。接着，可在反量化器的输出上执行反 DCT 524，其结果放置在帧存储 526 中，以在运动估计处理过程期间供访问。

**【0263】**根据本发明实施例的多媒体文件还能够包含许多心理视觉增强 528。心理视觉增强可能是基于人类视觉感觉的压缩视频的方法。这些技术将在下文进一步讨论，其一般地包含修改被量化器所使用的比特数，以表现视频的不同示象（aspect）。编码处理过程的其它方面还可包括心理视觉增强。

**【0264】**在一个实施例中，整个编码系统 500 可使用计算机来实现，该计算机被配置成执行上文所描述的各种各样的功能。这些功能的详细实现的例子，将在下文中提供。

### 3.5.1 预处理

**【0265】**由根据本发明实施例的编码器 500 选择性执行的预处理操作 502，其可使用许多信号处理技术，以提高编码视频质量。在一个实施例中，预处理 502 可包括下列的一个或全部：去隔行（deinterlacing）、时间/空间噪声抑制及尺寸重调。在所有这三个预处理技术都被使用的

实施例中，通常首先执行去隔行，然后执行时间/空间噪声抑制及尺寸重调。

### 3.5.2 运动估计和补偿

【0266】根据本发明实施例的视频编码器可依靠搜索在多个帧中重复的像素，来减少表现视频轨所需的像素数目。基本上，每个视频帧通常包含许多与前一帧相同的像素。编码器可进行几种类型的搜索，以匹配每个帧之间的像素（如宏块、像素、半像素，以及 1/4 像素），并在可能不降低图像质量的情况下去除那些冗余。使用运动估计，编码器可这样表现大多数的画面：仅仅通过记录自从最近一帧所发生的变化而不存储每帧的全部画面。在运动估计期间，编码器将其正在分析的帧划分成常被称为“macroblock（宏块）”的均匀网格块。对于帧中的每个“macroblock”，编码器可努力去发现在前帧中的匹配块。努力发现匹配块的处理过程被称为“运动搜索”。“macroblock”的运动可表现为 2 维矢量，例如，(x,y) 的表现形式。运动搜索算法可具有各种精确度地被执行。全像素搜索是一种搜索算法，在其中编码器将通过在任一维度上每次一个像素地按步通过引用帧来努力定位匹配的块。在半像素搜索中，编码器通过在任一维度上每次半个像素地按步通过引用帧来搜索匹配的块。编码器可使用 1/4 像素，其它像素分数或包括了大于一个像素的粒度的搜索。

【0267】图 3.5 所示的编码器实施例根据本发明实施例来执行运动估计。在运动估计期间，编码器访问预处理视频 502 和存储在帧存储 526 中的前一帧。通过对得到的量化器输出，执行反量化 522 和反 DCT 变换 524 产生前一帧。执行反函数的原因在于：处于帧存储中的帧和将在播放器根据本发明实施例解码它时出现的一样。

【0268】通过接受作为运动估计的结果所产生的块和矢量来执行运动补偿。该运动估计的结果是对编码的图像的近似，其可通过提供额外的纹理信息而与实际图像相匹配。

### 3.5.3 离散余弦变换



【0269】由图 3.5 所示的编码器执行的 DCT 和反 DCT 符合 ISO/IEC 14496-2:2001(E)，附录 A.1（编码变换）中规定的标准。

### 3.5.3.1 变换描述

【0270】DCT 是一种将一组空间域数据点变换到频域表现的方法。在视频压缩的情况下，2 维 DCT 将图像块转换为冗余更容易被利用的形式。频域块可以是稀疏矩阵，其可轻易通过熵编码被压缩。

### 3.5.3.2 对变换的心理视觉增强

【0271】可修改 DCT 系数，以通过降低区域内的量化噪声来提升被量化的图像的质量。所述区域是指对于人类观察者轻易可见的区域。另外，文件大小可通过增加图像中某些部分的量化噪声来减小，所述部分是人类观察者难于注意到的。

【0272】根据本发明实施例的编码器可执行被称为“慢速”心理视觉增强。该“慢速”心理视觉增强分析视频图像的块，并决定在此允许一些噪声是否能够在不降低视频外观的情况下节省一些比特。每块的处理使用一种度量。该处理被称为“慢速”处理，这是因为其执行了相当数量的计算，以避免阻塞或伪响铃。

【0273】根据本发明实施例的编码器的其它实施例实现了“快速”心理视觉增强。该“快速”心理视觉增强可控制噪声在块内出现的位置，并可以调整量化噪声。

【0274】“慢速”和“快速”心理视觉增强都将在下文中非常详细地讨论。其它心理视觉增强可根据本发明实施例地被执行，包括下述增强：控制在图像边缘处的噪声以及力求将较高等级的量化噪声集中在对人类视觉不易见到的图像区域。

### 3.5.3.3 “慢速”心理视觉增强

【0275】“慢速”心理视觉增强对视频图像块进行分析，并确定是否允许一些噪声就能够在不降低视频外观的情况下节省比特。在一个实施例中，算法包括两个阶段。第一阶段包括产生输入照度像素的差异图像。该差异图像将以下文所述的方式来产生。第二阶段包括在量化之前修改 DCT 系数。

### 3.5.3.3.1 产生差异图像

【0276】从未压缩的源像素  $p_{xy}$  来计算每个差异图像的像素  $p'_{xy}$  如下：

$$p'_{xy} = \max(|p_{x+1,y} - p_{xy}|, |p_{x-1,y} - p_{xy}|, |p_{xy+1} - p_{xy}|, |p_{xy-1} - p_{xy}|)$$

其中  $p'_{xy}$  将在 0 到 255 的范围内（假定为 8 比特视频）。

### 3.5.3.3.2 DCT系数修改

【0277】对 DCT 系数的修改可包括计算块响铃因数（ringing factor）、块能量计算，以及系数值的实际修改。

### 3.5.3.3.3 计算块响铃因数

【0278】对于图像的每个块，“响铃因数”可基于差异图像的局部区域来计算。在块被定义为  $8 \times 8$  块的实施例中，响铃因数可使用下述方法确定。

【0279】首先，基于  $8 \times 8$  块之内的最大和最小照度像素值确定一个阈值：

$$threshold_{block} = floor((max_{block} - min_{block})/8) + 2$$

【0280】差异图像和阈值用来产生一个块的相邻区域中的“平坦(Flat)”像素的映射。每个块具有不同阈值的可能性阻止了整个帧的平坦像素映射的创建。该映射产生如下：

$$\begin{aligned} flat_{xy} &= 1, & \text{当 } p'_{xy} < threshold_{block} \\ flat_{xy} &= 0, & \text{其它情况} \end{aligned}$$

【0281】平坦像素的映射可根据一简单逻辑操作而被滤波：

$$\begin{aligned} flat'_{xy} &= 1, & \text{当 } flat_{xy} = 1 \text{ 且 } flat_{x-1,y} = 1 \text{ 且 } flat_{xy-1} = 1 \text{ 且 } flat_{x-1,y-1} = 1 \\ flat'_{xy} & & \text{其它情况} \end{aligned}$$

【0282】滤波映射中的平坦像素接着沿一个覆盖了该  $8 \times 8$  块的  $9 \times 9$  的区域被记数。

$$flatcount_{block} = \sum flat'_{xy} \quad \text{对于 } 0 \leq x < 8, \text{ 且 } 0 \leq y < 8$$

【0283】可使用下述表达式来估计可见的伪响铃的风险：

$$ringingbrisk_{block} = ((flatcount_{block} - 10) \times 256 + 20) / 40$$

【0284】使用下述表达式，可接下来获得该  $8 \times 8$  块的响铃因数：

$$\begin{aligned} \text{Ringingfactor} &= 0 & \text{ringingrisk} > 255 \\ &= 255 & \text{ringingrisk} < 0 \\ &= 255 - \text{ringingrisk} & \text{其它情况} \end{aligned}$$

#### 3.5.3.3.4 计算块能量

【0285】使用下面的程序可计算图像块的能量。在若干个实施例中，使用图像的  $8 \times 8$  块。

【0286】前向 DCT 在源图像上被执行：

$$T = fDCT(S)$$

【0287】其中  $S$  是所讨论的  $8 \times 8$  块的 64 个源图像照度值，以及  $T$  是变换版本的源图像的同一部分。

【0288】在特定系数位置处的能量可定义为那个系数值的平方：

$$e_k = t_k^2 \quad \text{对于 } 0 = k = 63$$

【0289】其中  $t_k$  是变换块  $T$  的第  $k$  个系数。

#### 3.5.3.3.5 系数修改

【0290】DCT 系数的修改可根据下列处理过程来执行。在若干个实施例中，处理在量化前对每个非零 AC DCT 系数执行。每个系数的大小会变化一个小  $\Delta$ ，该  $\Delta$  的数值根据心理视觉技术来确定。

【0291】对每个非零 AC 系数  $c_k$  的 DCT 系数修改的执行，依靠基于局部和块能量的能量计算，其使用了下列公式：

$$\text{energy}_k = \max(a_k \times e_k, 0.12 \times \text{totalenergy})$$

【0292】其中  $a_k$  是常数，其数值依赖于系数位置，如下表所示：

表 3 系数表

0.0	1.0	1.5	2.0	2.0	2.0	2.0	2.0
1.0	1.5	2.0	2.0	2.0	2.0	2.0	2.0
1.5	2.0	2.0	2.0	2.0	2.0	2.0	2.0
2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0
2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0
2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0
2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0
2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0

【0293】可利用如下公式根据该块的响铃因数修改能量：

$$energy'_k = ringingfactor \times energy_k$$

【0294】该结果值在被作为查找表(LUT)的输入之前，被移位和限幅。

$$e_k = \min(1023, 4 \times energy'_k)$$

$$d_k = LUT_i \quad \text{其中 } i = e_k$$

【0295】该查找表如下被计算：

$$LUT_i = \min(\text{floor}(k_{texture} \times ((i + 0.5) / 4)^{1/2} + k_{flat} \times offset), 2 \times Q_p)$$

【0296】依赖于量化因子  $Q_p$  的数值 “offset” 可描述如下表：

表 4 作为  $Q_p$  值函数的 offset

$Q_p$	offset	$Q_p$	offset
1	-0.5	16	8.5
2	1.5	17	7.5
3	1.0	18	9.5
4	2.5	19	8.5
5	1.5	20	10.5
6	3.5	21	9.5
7	2.5	22	11.5
8	4.5	23	10.5
9	3.5	24	12.5
10	5.5	25	11.5
11	4.5	26	13.5
12	6.5	27	12.5
13	5.5	28	14.5

14	7.5	29	13.5
15	6.5	30	15.5
		31	14.5

【0297】变量  $k_{texture}$  和  $k_{flat}$  分别控制着平坦和纹理化区域的心理视觉作用的强度。在一个实施例中，它们具有在 0 到 1 范围内的值，其中 0 代表没有作用，而 1 意味着全作用。在一个实施例中， $k_{texture}$  和  $k_{flat}$  的数值被如下建立：

照度：

$$k_{texture} = 1.0$$

$$k_{flat} = 1.0$$

色度：

$$k_{texture} = 1.0$$

$$k_{flat} = 0.0$$

【0298】查找表的输出 ( $d_k$ ) 被用来通过额外的处理，来修改 DCT 系数的大小：

$$c'_k = c_k - \min(d_k, |c_k|) \times \text{sgn}(c_k)$$

【0299】最终，DCT 系数  $c_k$  由修改后的系数  $c'_k$  所代替，并向前传递以供量化。

#### 3.5.3.4 “快速”心理视觉增强

【0300】“快速”心理视觉增强可被在 DCT 系数上执行，这通过计算输入明亮像素的“重要性”映射，然后修改 DCT 系数。

##### 3.5.3.4.1 计算“重要性”映射

【0301】可通过计算输入视频帧的明亮位置上的每个像素的“重要性”数值，来产生“重要性”映射。在若干个实施例中，“重要性”数值近似为人眼对位于特定像素的任何失真的灵敏度。“重要性”映射是一列像素“重要性”数值。

【0302】可首先通过计算围绕着像素 ( $d_{xy}$ ) 的多个像素的块的动态范围，来确定该像素的“重要性”。在若干个实施例中，一个以像素位置

$(x,y)$  为中心的  $3 \times 3$  像素块的动态范围可如下计算：将区域中最亮的像素数值减去区域中最暗的像素数值。

【0303】能够根据像素的动态范围获得像素的“重要性” ( $m_{xy}$ )，如下：

$$m_{xy} = 0.08 / \max(d_{xy}, 3) + 0.001$$

#### 3.5.3.4.2 修改DCT系数

【0304】在一个实施例中，DCT 系数的修改包括基函数能量矩阵和  $\Delta$  查找表的产生。

#### 3.5.3.4.3 基函数能量矩阵的产生

【0305】一组基函数能量矩阵可被用来修改 DCT 系数。这些矩阵包含了常数值，其在编码前被计算。 $8 \times 8$  矩阵被用于 64 个 DCT 基函数的每一个。每个矩阵描述了  $8 \times 8$  块中的每个像素是如何受到其相关系数的修改的影响的。第  $k$  个基函数能量矩阵由此获得：一  $8 \times 8$  矩阵  $A_k$ ，其相关系数设置为 100，其它系数设置为 0。

$$\begin{aligned} a_{kn} &= 100 && \text{如果 } n=k \\ &= 0 && \text{其它情况} \end{aligned}$$

其中

【0306】 $n$  代表系数在  $8 \times 8$  矩阵中的位置； $0=n=63$

【0307】在矩阵上执行反 DCT，以得到进一步的  $8 \times 8$  矩阵  $A'_k$ 。矩阵的元素 ( $a'_{kn}$ ) 代表了第  $k$  个 DCT 基函数。

$$A'_k = iDCT(A_k)$$

【0308】接着，对变换矩阵中的每个数值取平方：

$$b_{kn} = a'^2_{kn} \quad \text{对于 } 0=n=63$$

【0309】该处理被执行 64 次，以产生基函数能量矩阵  $B_k$ ， $0=k=63$ ，每一个包含了 64 个自然数。每个矩阵的值是一种度量，其度量了  $8 \times 8$  块中第  $n$  位置的像素将受到系数  $k$  的任何误差或修改多大地影响。

#### 3.5.3.4.4 $\Delta$ 查找表的产生

【0310】查找表 (*LUT*) 能够用来加快系数修改  $\Delta$  的计算。表的内容可以这样的方式产生：其依赖于所需的“快速”心理视觉增强的强度和量化参数 ( $Q_p$ )。

【0311】查找表的数值可根据下述关系产生：

$$LUT_i = \min(\text{floor}(128 \times k_{\text{texture}} \times \text{strength} / (i + 0.5) + k_{\text{flat}} \times \text{offset} + 0.5), 2 \times Q_p)$$

其中

【0312】 $i$  是在查找表内的位置， $0=i=1023$ 。

*Strength* 和 *offset* 依赖于量化因子或量化参数  $Q_p$ ，如下表所示：

表 5 *Strength* 和 *offset* 的值和  $Q_p$  值之间的关系

$Q_p$	<i>strength</i>	<i>offset</i>	$Q_p$	<i>strength</i>	<i>offset</i>
1	0.2	-0.5	16	2.0	8.5
2	0.6	1.5	17	2.0	7.5
3	1.0	1.0	18	2.0	9.5
4	1.2	2.5	19	2.0	8.5
5	1.3	1.5	20	2.0	10.5
6	1.4	3.5	21	2.0	9.5
7	1.6	2.5	22	2.0	11.5
8	1.8	4.5	23	2.0	10.5
9	2.0	3.5	24	2.0	12.5
10	2.0	5.5	25	2.0	11.5
11	2.0	4.5	26	2.0	13.5
12	2.0	6.5	27	2.0	12.5
13	2.0	5.5	28	2.0	14.5
14	2.0	7.5	29	2.0	13.5
15	2.0	6.5	30	2.0	15.5
			31	2.0	14.5

【0313】变量  $k_{\text{texture}}$  和  $k_{\text{flat}}$  分别控制着平坦和纹理化区域的心理视觉作用的强度。在一个实施例中，它们具有在 0 到 1 范围内的值，其中 0

代表没有作用，而 1 意味着全作用。在一个实施例中， $k_{texture}$  和  $k_{flat}$  的数值被如下建立：

照度：

$$\begin{aligned}k_{texture} &= 1.0 \\k_{flat} &= 1.0\end{aligned}$$

色度：

$$\begin{aligned}k_{texture} &= 1.0 \\k_{flat} &= 0.0\end{aligned}$$

#### 3.5.3.4.5 DCT系数的修改

【0314】可使用上面计算的数值来修改 DCT 系数。在一个实施例中，在量化前按照下列处理过程来修改每个非零 AC DCT 系数。

【0315】开始“能量”值 ( $e_k$ ) 如此计算：相关基函数能量矩阵和来自重要性映射的合适的  $8 \times 8$  块的点积。这里的“能量”是一个度量，其度量了在特定系数处的量化误差是如何被人类观察者所感知的。它是像素重要性和像素基函数能量的乘积之和。

$$e_k = M \cdot B_k$$

其中

【0316】 $M$  包含了  $8 \times 8$  块的重要性映射数值；而  $B_k$  是第  $k$  个基函数能量矩阵。

【0317】结果的“能量”值，在被用作  $\Delta$  查找表的索引 ( $d_k$ ) 之前，移位和限幅。

$$\begin{aligned}e'_k &= \min[1023, \text{floor}(e_k / 32768)] \\d_k &= LUT_i\end{aligned}$$

其中

$$i = e'_k$$

【0318】通过额外处理， $\Delta$  查找表的输出可被用来修改 DCT 系数的大小：



$$c'_k = c_k - \min(d_k, |c_k|) \times \text{sign}(c_k)$$

【0319】DCT 系数  $c_k$  由修改后的系数  $c'_k$  所代替, 并向前传递以供量化。

### 3.5.4 量化

【0320】根据本发明实施例的编码器可使用标准量化因子, 例如由国际电信联盟在 Video Coding for Low Bitrate Communication, ITU-T Recommendation H.263, 1996 中所定义的量化因子。

#### 3.5.4.1 量化的心理视觉增强

【0321】根据本发明实施例的一些编码器, 使用了心理视觉增强, 其利用了人类视觉的心理学效应, 以获得更高效的压缩。该心理视觉效应可被应用在帧级和宏块级。

#### 3.5.4.2 帧等级的心理视觉增强

【0322】当增强在帧级应用时, 其作为速率控制算法的一部分, 并且它的目的是调整编码以最佳地使用给定数量的比特率, 以保证人眼感觉到最好的视觉质量。帧率心理视觉增强由下述理论激发: 人类视觉在高速运动时趋于忽略细节, 而在图像静止时人类视觉又趋于注意细节。在一个实施例中, 运动的量通过考虑帧的绝对差之和 (SAD) 来确定。在一个实施例中, SAD 值通过对两块在所排列照度像素差异的绝对值取和来确定。在若干个实施例中, 使用了  $16 \times 16$  像素块的绝对差。在处理分数像素偏移的实施例中, 在绝对差之和计算之前, 执行 MPEG-4 标准 (ISO/IEC 标准, 其由 ISO/IEC 的运动图像专家组所开发) 中规定的插值。

【0323】帧级心理视觉增强仅应用于视频轨的 P 帧, 其基于帧的 SAD 值。在编码期间, 心理视觉模块保存了视频轨的所有 P 帧的平均 SAD (即,  $\overline{SAD}$ ) 和每帧 SAD 到它们总 SAD 之间平均距离 (即  $\overline{DSAD}$ ) 的记录。可使用指数移动平均值算法完成该取平均。在一个实施例中, 上文所描述的一遍速率控制算法可被用在取平均期间 (见上文描述)。

【0324】对于所编码视频轨的每个 P 帧，将有心理视觉校正被施加到帧量化因子  $Q$ （从速率控制模块获得）。在一个实施例中，该处理包括使用下述公式计算比率  $R$ ：

$$R = \frac{SAD - \overline{SAD}}{DSAD} - I$$

【0325】其中

$I$  是常数，其经常设置为 0.5。 $R$  被限幅在区间[-1,1]之内。

【0326】接下来，量化因子根据比率  $R$  来调整，其通过下面的计算：

$$Q_{adj} = Q \lfloor Q(1 + R \cdot S_{frame}) \rfloor$$

其中

【0327】 $S_{frame}$  是帧级心理视觉增强的强度常数。

【0328】 $S_{frame}$  常数确定了多大强度的调整可被用在帧级心理视觉。在一个编解码器的实施例中，将  $S_{frame}$  设置为 0.2、0.3 或 0.4 的选项是允许的。

#### 3.5.4.3 宏块级的心理视觉增强

【0329】根据本发明实施例的、在宏块级使用心理视觉增强的编码器尝试标识出对于人类观察者来说对视频的视觉质量显著的宏块，所示编码器还尝试以较高质量对这些宏块编码。宏块级心理视觉增强的作用是这样的：其将比特从帧的不太重要的部分移走，并将其施加到帧中更为重要的部分。在若干个实施例中，使用了三种技术来获得增强。所述的三个技术基于平滑度（smoothness）、亮度（brightness）和宏块 SAD。在其它实施例中，可单一使用所述技术中的任何一个，或联合所述技术中的另一种技术使用，或完全使用另一种技术。

【0330】在一个实施例中，所有上文所描述的三个宏块级心理视觉增强共享一个共同的参数， $S_{MB}$ ，其控制着宏块级心理视觉增强的强度。接下来，通过下面所述的计算，从这一强度参数和帧量化因子  $Q_{frame}$  可得出宏块的最大和最小量化因子：

$$Q_{MBMax} = \frac{Q_{frame}}{(1 - S_{MB})},$$

以及  $Q_{MBMin} = Q_{frame} \cdot (1 - S_{MB})$

【0331】其中

$Q_{MBMax}$  是最大量化因子;

$Q_{MBMin}$  是最小量化因子;

【0332】数值  $Q_{MBMax}$  和  $Q_{MBmin}$  定义了整个帧宏块量化因子的上界和下界。在一个实施例中, 将值  $S_{MB}$  设置为 0.2、0.3 或 0.4 中的任何一个的选项是允许的。在其它实施例中, 可使用其它数值用于  $S_{MB}$ 。

#### 3.5.4.3.1 亮度增强

【0333】在心理视觉增强是基于宏块的亮度来执行的实施例中, 编码器尝试将亮度较大的宏块以较高质量编码。这种增强的理论基础是: 帧中相对暗的部分或多或少会被人类观察者所忽略。这种宏块心理视觉增强被施加到视频轨的 I 帧和 P 帧。对于每一帧, 编码器首先浏览整个帧。平均亮度 ( $\overline{BR}$ ) 被计算, 并且到该平均值的亮度差异平均 ( $\overline{DBR}$ ) 也被计算。接着, 这些数值被用来产生出两个阈值 ( $T_{BRLower}$ ,  $T_{BRUpper}$ ), 该阈值用来指示心理视觉增强是否被应用:

$$T_{BRLower} = \overline{BR} - \overline{DBR}$$

$$T_{BRUpper} = \overline{BR} + (\overline{BR} - T_{BRLower})$$

【0334】接着, 基于所述两个阈值施加亮度增强, 使用下面规定的条件来产生所希望的宏块量化因子 ( $Q_{MB}$ )

$$Q_{MB} = Q_{MBMin} \quad \text{当 } BR > T_{BRUpper}$$

$$Q_{MB} = Q_{frame} \quad \text{当 } T_{BRLower} \leq BR \leq T_{BRUpper}, \quad \text{及}$$

$$Q_{MB} = Q_{MBMax} \quad \text{当 } BR < T_{BRLower}$$

【0335】其中,

$BR$  是特定宏块的亮度值。

【0336】在编码器与 MPEG-4 标准兼容的实施例中, 宏块级心理视觉亮度增强技术对从一个宏块到下一个宏块的量化因子改变, 不能超过  $\pm 2$ 。因此, 所计算的  $Q_{MB}$  可需要基于前面宏块所用的量化因子来改进。

### 3.5.4.3.2 平滑度增强

【0337】根据本发明实施例的、包含平滑度心理视觉增强的编码器基于被编码图像的空间变化，改进量化因子。平滑度心理视觉增强的使用可由下述理论来激发：人类视觉对图像平滑部分中的量化误差具有增加的灵敏度。因此，平滑度心理视觉增强能够包含增加表现图像较平滑部分的比特数目，并且减少了图像中高度空间变化部分的比特数目。

【0338】在一个实施例中，图像某部分的平滑度可度量为宏块中像素的照度和宏块亮度的平均差异 ( $\overline{DR}$ )。图 3.6 显示了一种方法，其根据本发明实施例在 I 帧上执行平滑度心理视觉增强。处理过程 540，包括检查整个帧以计算 542  $\overline{DR}$ 。然后，用于施加平滑度增强的阈值， $T_{DR}$ ，可使用下列计算获得 544：

$$T_{DR} = \frac{\overline{DR}}{2}$$

【0339】下面的平滑度增强基于该阈值来执行 546。

$$\begin{aligned} Q_{MB} &= Q_{frame} && \text{当 } DR \geq T_{BRUpper}, \text{ 及} \\ Q_{MB} &= Q_{MBMin} && \text{当 } DR < T_{DR} \end{aligned}$$

【0340】其中

$Q_{MB}$  是所希望的宏块量化因子

$DR$  是宏块的偏离值（即，平均照度-平均亮度）

【0341】根据 MPEG-4 标准来编码文件的实施例受到如上所述的限制：宏块级的量化因子从一个宏块到下一个宏块可最多在  $\pm 2$  变化。

### 3.5.4.3.3 宏块 SAD 增强

【0342】根据本发明实施例的编码器可使用宏块 SAD 心理视觉增强。宏块 SAD 心理视觉增强能够用来增加静态宏块的细节，并允许减少帧的用于高速运动场景的部分的细节。

【0343】图 3.7 示出了根据本发明实施例的宏块 SAD 心理视觉增强的执行过程。处理过程 570 包括检查 572 整个 I 帧，以确定对于整个帧的

所有宏块的平均 SAD (即,  $\overline{MBSAD}$ ), 并且还获得了某一宏块的 SAD 相对于该平均值的平均差异 (即  $\overline{DMBSAD}$ )。在一个实施例中, 所有这些宏块将在交互帧 (inter-frame) 编码的宏块取平均 (即, 所编码的宏块依赖于前面的编码视频帧来使用运动补偿或其它补偿)。接下来, 用于宏块 SAD 增强的两个阈值从这些平均中产生, 其使用了下面的公式:

$$T_{MBSADLower} = \overline{MBSAD} - \overline{DMBSAD}, \text{ 及}$$

$$T_{MBSADUpper} = \overline{MBSAD} + \overline{DMBSAD}$$

【0344】其中,

$T_{MADLower}$  是下阈值

$T_{MBSADUpper}$  是上阈值, 其如果必要, 则被限定小于 1024。

【0345】然后, 宏块 SAD 增强基于上述两个阈值, 并根据下述条件被应用 576:

$$Q_{MB} = Q_{MBMax} \quad \text{当 } MBSAD > T_{MBSADUpper},$$

$$Q_{MB} = Q_{frame} \quad \text{当 } T_{MADLower} \leq MBSAD \leq T_{MBSADUpper}$$

$$Q_{MB} = Q_{MBMin} \quad \text{当 } MBSAD < T_{MBSADLower}$$

【0346】其中

$Q_{MB}$  宏块所希望的量化因子

$MBSAD$  是特定宏块的 SAD 值

【0347】根据 MPEG-4 规范来编码文件的实施例受到如上所述的限制: 宏块级别的量化因子从一个宏块到下一个宏块可最多在  $\pm 2$  变化。

### 3.5.5 速率控制

【0348】由根据本发明实施例的编码器所使用的速率控制技术可确定编码器如何使用所分配的比特率来编码视频序列。编码器将通常力求编码到预定的比特率, 而速率控制技术负责使由编码器所产生的比特率尽可能接近地匹配预定比特率。速率控制技术还力求以这样的方式来分配比特率, 该方式确保将当视频序列被解码时, 具有最高的视觉质量。通过调整量化因子来执行大量速率控制。量化因子确定了编码器编码视频序列有多好。较小的量化因子将导致较高的质量和较高的

比特消耗。因此，速率控制算法力求以这样的方式来改进量化因子：平衡视频质量和比特消耗之间的竞争利益。

**【0349】**根据本发明实施例的编码器可应用多种不同的速率控制技术中的任意一种。在一个实施例中，使用了单遍速率控制技术。在另一个实施例中，使用了两次（或多次）通过速率控制技术。另外，可根据需要执行“视频缓冲验证”速率控制。这些技术的特定例子将在下文讨论。然而根据本发明的实施例，可在编码器中使用任何速率控制技术。

#### 3.5.5.1 一遍速率控制

**【0350】**根据本发明实施例的一遍速率控制技术的实施例力求允许用于高速运动场景的高比特率峰。在若干个实施例中，一遍速率控制技术力求响应于场景中运动量的增加而缓慢地增加比特率，以及响应于场景中运动的缩减而迅速地减少比特率。

**【0351】**在一个实施例中，一遍速率控制算法使用两种平均周期来跟踪比特率。长时平均用以确保总比特率收敛，而短时平均用以能够响应于场景中运动量的变化。

**【0352】**图 3.8 示出了根据本发明实施例的一遍速率控制技术。一遍速率控制技术 580 通过初始化 584 编码器而着手 582，以所需比特率、视频帧率和多种其它参数（将进一步在下文讨论）来进行所述初始化。如果帧需要量化 586，则重新获得 588 浮点变量，并通过四舍五入该浮点变量到最近的整数来获得量化因子。接下来，帧被编码 590。在帧编码期间进行观察，以能够确定 592 新的量化因子值。该处理过程决定 594 重复，直到没有更多的帧。在没有更多帧时，编码完成 596。

**【0353】**如上所述，编码器通过多种参数而被初始化 584。这些参数是“比特率”、“帧率”、“最大关键帧间隔”、“最大量化因子”、“最小量化因子”、“平均周期”、“响应时间”，以及“升降比（down/up ratio）”。

##### 3.5.5.1.1 “比特率”

**【0354】**“比特率”参数设置了编码的目标比特率。

##### 3.5.5.1.2 “帧率”

【0355】“帧率”参数定义了视频帧之间的周期。

#### 3.5.5.1.3 “最大关键帧间隔”

【0356】“最大关键帧间隔”指明了关键帧之间的最大间隔。当编解码器检测到场景变化，则关键帧通常就自动地被插入到编码的视频中。对于没有一个单一切口的长间隔的连续场景的环境中，关键帧可被插入，以确保关键帧之间的间隔总是小于或等于“最大关键帧间隔”。在一个实施例中，“最大关键帧间隔”参数能够被设置为 300 帧的数值。在其它实施例中，可使用其它数值。

#### 3.5.5.1.4 “最大量化因子”和“最小量化因子”

【0357】“最大量化因子”和“最小量化因子”参数设置了使用在编码中的量化因子的上界和下界。在一个实施例中，量化因子区间能够被设置在 1 到 31 之间的数值。

#### 3.5.5.1.5 “平均周期”

【0358】“平均周期”参数控制在改进量化因子时所考虑的视频的数量。较长的平均周期将通常导致编码的视频具有较准确的总速率。在一个实施例中，使用了“平均周期”2000。尽管在其它实施例中，可使用其它数值。

#### 3.5.5.1.6 “响应时间”

【0359】“响应时间”参数确定了对于最近场景中运动的变化，编码器适应得有多快。较长的“响应时间”值可导致较好质量的高速运动场景，以及较差质量的低运动场景。在一个实施例中，使用为 10 的“响应时间”。尽管在其它实施例中，可使用其它数值。

#### 3.5.5.1.7 “升降比”

【0360】“升降比”参数控制了量化因子调整的相对灵敏度，该量化因子的调整作为对高或低运动场景的反应。较大的值通常导致较高质量高速运动场景，并增加了比特率消耗。在一个实施例中，使用为 20 的“升降比”。尽管在其它实施例中，可使用其它数值。

#### 3.5.5.1.8 计算量化因子值

【0361】如上面所讨论的，一遍速率控制技术包括在每个帧编码之后，计算量化因子值。下文是对根据本发明实施例的技术的讨论，该技术可被用于更新量化因子值。

【0362】编码器维持两个指数移动平均值具有与“平均周期”( $P_{average}$ )和“响应时间”( $P_{reaction}$ )相等的周期，比特率的移动平均值。该两个指数移动平均值可根据关系来计算。

$$A_t = A_{t-1} \cdot \frac{P-T}{P} + B \cdot \frac{T}{P}$$

【0363】其中，

$A_t$ 是在时刻  $t$  的平均值；

$A_{t-1}$ 是在时刻  $t-T$  的平均值（通常在前面帧的平均值）；

$T$ 代表间隔周期（通常帧时间）；和

$P$ 是平均周期，其可以是  $P_{average}$  或  $P_{reaction}$ 。

【0364】接着，上面所计算的移动平均通过以在视频的当前时刻和最后时刻之间的时间间隔进行分割，而被调整到比特率，其使用了下面的计算：

$$R_t = A_t \frac{1}{T}$$

【0365】其中

$R_t$ 是比特率；

$A_t$ 是移动平均的任何一个；以及

【0366】 $T$ 是在当前时刻和最后时刻之间的时间间隔（它通常是帧率的倒数）。

【0367】编码器可计算下一帧的目标比特率 ( $R_{target}$ ) 如下：

$$R_{target} = R_{overall} + (R_{overall} - R_{average})$$

【0368】其中

$R_{overall}$ 是整体视频的总比特率；以及

$R_{average}$ 是使用长平均周期的平均比特率。



【0369】在若干个实施例中，目标比特率被下界限定到总比特率的 75%。如果目标比特率低于这个界，则它将被强制等于该界以确保视频的质量。

【0370】接着，编码器基于  $R_{target}$  和  $R_{reaction}$  之间的差异，来更新内部量化因子。如果  $R_{reaction}$  小于  $R_{target}$ ，则具有前帧有相对低复杂度的可能性。因此，量化因子可通过执行下述计算而被减小：

$$Q'_{internal} = Q_{internal} \left( 1 - \frac{1}{P_{reaction}} \right)$$

【0371】当  $R_{reaction}$  大于  $R_{target}$  时，则具有前帧有相对高级别复杂度的、相当大的可能性。因此，量化因子可通过执行下述计算而被增加：

$$Q'_{internal} = Q_{internal} \left( 1 + \frac{1}{SP_{reaction}} \right)$$

【0372】其中

S 是所述“升降比”。

#### 3.5.5.1.9 B-VOP编码

【0373】上面所描述的算法还可被应用到 B-VOP 编码。当 B-VOP 可以在编码中时，基于跟随 B-VOP 之后的 B-VOP 的量化因子 ( $Q_p$ ) 来选择 B-VOP 的量化因子 ( $Q_B$ )。该数值可根据下面关系而获得：

$$Q_B = 2 \cdot Q_p \quad \text{对于 } Q_p \leq 4$$

$$Q_B = 5 + \frac{3}{4} Q_p \quad \text{对于 } 4 < Q_p \leq 20$$

$$Q_B = Q_p \quad \text{对于 } Q_p \geq 20$$

#### 3.5.5.2 两遍速率控制

【0374】根据本发明实施例的、使用两遍（或多遍）速率控制技术的编码器，可确定视频序列在第一遍时的特性，然后根据对整个序列特性的了解，编码该视频序列。因此，编码器可基于每帧相比视频序列中其它帧的相对复杂度，来调整每帧的量化水平。

【0375】根据本发明实施例的两遍速率控制技术，编码器执行第一遍，在其中根据上文所描述的一遍速率控制技术编码视频，并记录每帧的复杂度（可使用多种不同的用于复杂度度量的量度中的任意一种）。因此，平均复杂度和平均量化因子（ $Q_{ref}$ ）可基于第一遍而被确定。在第二遍中，以基于第一遍期间所计算的复杂度数值而确定的量化因子，来编码比特流。

#### 3.5.5.2.1 I-VOPs的量化因子

【0376】用于 I-VOPs 的量化因子  $Q$ ，在下一帧不是 I-VOPs 时被设置为  $0.75 \times Q_{ref}$ 。如果下一帧是 I-VOPs，则  $Q$ （当前帧的）被设置为  $1.25 \times Q_{ref}$ 。

#### 3.5.5.2.2 P-VOPs的量化因子

【0377】P-VOPs 的量化因子可以使用下面的表达式来确定：

$$Q = F^{-1} \left\{ F(Q_{ref}) \left( \frac{\overline{C_{complexity}}}{C_{complexity}} \right)^k \right\}$$

【0378】其中

$C_{complexity}$  是帧的复杂度；

$\overline{C_{complexity}}$  是视频序列的平均复杂度；

【0379】 $F(x)$ 是函数，其提供了一个帧复杂度必须与之相乘的数，以给出比特数目，该比特数目是使用具有量化数值  $x$  的量化因子来编码帧所需要的；

$F^{-1}(x)$ 是  $F(x)$ 的反函数；以及

$k$  是强度参数。

【0380】下表定义了函数  $F(Q)$  的实施例，其可被用来产生帧复杂度必须与之相乘的因数，以确定使用具有量化因子  $Q$  的编码器来编码帧所需的比特数。

表 6 关于  $Q$  的  $F(Q)$  的数值

$Q$	$F(Q)$	$Q$	$F(Q)$
1	1	9	0.013
2	0.4	10	0.01
3	0.15	11	0.008
4	0.08	12	0.0065
5	0.05	13	0.005
6	0.032	14	0.0038
7	0.022	15	0.0028
8	0.017	16	0.002

【0381】如果强度参数  $k$  被选为 0，则结果是一个常数量化因子。当强度参数被选为 1 时，量化因子与  $C_{complexity}$  成比例。根据本发明实施例的几个编码器具有等于 0.5 的强度参数。

#### 3.5.5.2.3 B-VOPs的量化因子

【0382】可使用与上文描述的在一遍技术中选择 B-VOPs 量化因子的相同技术，来选择 B-VOPs 的量化因子  $Q$ 。

#### 3.5.5.3 视频缓冲验证速率控制

【0383】表现一帧所需的比特数可依赖视频序列的特点而变化。大多数的通信系统在恒定比特率下操作。可变比特率通信可能遇到的问题是分配足够的资源以掌握资源使用的峰。根据本发明的几个实施例以这样的目的来编码视频：当可变比特率通信的比特率达到峰值时，阻止解码器视频缓冲器的上溢。

【0384】视频缓冲验证器（VBV）速率控制的目标可包括产生视频，其在传输时，将不会超过解码器的缓冲器。另外，编码的视频匹配一目标比特率以及该速率控制产生高质量视频也是需要的。

【0385】根据本发明几个实施例的解码器提供了至少两个 VBV 速率控制技术的选择。VBV 速率控制技术之一称为因果速率控制，而其它技术称为第 N 遍速率控制。

##### 3.5.5.3.1 因果速率控制

【0386】因果 VBV 速率控制可与一遍速率控制技术协力使用，并产生简单基于当前和前面量化因子数值的输出。

【0387】根据本发明实施例的编码器包括因果速率控制，其包括根据下面关系，为帧  $n$  设置量化因子（即  $Q_n$ ）。

$$\frac{1}{Q'_n} = \frac{1}{Q'_{n-1}} + X_{\text{bitrate}} + X_{\text{velocity}} + X_{\text{size}}$$

$$\frac{1}{Q_n} = \frac{1}{Q'_n} + X_{\text{drift}}$$

【0388】其中

$Q'_n$  是由单遍速率控制所估计的量化因子；

$X_{\text{bitrate}}$  通过确定目标比特率而被计算，确定目标比特率是基于从所需比特率的偏离；

$X_{\text{velocity}}$  基于直到 VBV 缓冲器上溢或下溢的估计时间而被计算；

$X_{\text{size}}$  仅被应用在 P-VOPs 的结果上，并且基于压缩的 P-VOPs 的大小随时间变化的速率而被计算；

$X_{\text{drift}}$  是从所需比特率的偏移。

【0389】在若干个实施例中，因果 VBV 速率控制可被强制丢弃帧及插入填充 (stuffing) 以遵守 VBV 模型。如果压缩的帧突然包含了过多或过少的比特，则它可被丢弃或填充。

#### 3.5.5.3.2 第N遍VBV速率控制

【0390】第  $N$  遍 VBV 速率控制可与多遍速率控制技术协力使用，并且它使用了在前面分析视频序列期间所储存的信息。根据本发明几个实施例的编码器执行  $N$  遍 VBV 速率控制，其根据图 3.9 中所示的处理过程。处理过程 600 从第一遍着手，其执行分析 (602)。执行映射产生 (604) 以及策略产生 (606)。接着，执行第  $N$  遍 VBV 速率控制 (608)。

#### 3.5.5.3.3 分析

【0391】在一个实施例中，一遍使用一些因果速率控制的方式，并且记录每一帧的数据，所述数据关于这样的事物：帧持续时间、帧编码类型、所用的量化因子、所产生的运动比特，以及所产生的纹理比特。另外，也可记录全局信息，例如时间尺度、分辨率以及编解码器设置。

#### 3.5.5.3.4 映射产生

【0392】来自分析的信息可被用来产生视频序列的映射。该映射可为每一帧（I/B/P）指定编码类型，并且可包括关于帧持续时间、运动复杂度，以及纹理复杂度的每帧的数据。在其它实施例中，映射还能够包含信息，其使得能够更好的预测量化因子及其它参数对压缩的帧大小和感觉失真的影响。在若干个实施例中，在第 N-1 遍完成后，执行映射产生。

#### 3.5.5.3.5 策略产生

【0393】映射可被用来设计一种策略，其关于第 N 遍速率控制将如何被执行。在每帧被编码后的 VBV 缓冲器的理想级别可被设计。在一个实施例中，策略产生导致每一帧的信息包含了所需的压缩帧大小、估计的帧量化因子。在若干个实施例中，在映射产生之后，第 N 遍之前，执行策略产生。

【0394】在一个实施例中，策略产生处理过程包括使用迭代处理以仿真编码器并确定每一帧所需的量化因子值，其依靠了设法保持该量化因子与中值量化因子值尽可能的接近。可使用二进制搜索以产生整个视频序列的基础量化因子。该基础量化因子是常数值，其引起仿真器获得所需的目标比特率。一旦找到基础量化因子，策略产生处理就将包括对 VBV 约束的考虑。在一个实施例中，使用常数量化因子，如果这不会更改 VBV 约束。在其它实施例中，量化因子基于视频帧中的运动复杂度而被调节。这可进一步扩展到来自场景变化和其它时域效应的合并掩蔽（incorporate masking）

#### 3.5.5.3.6 内环路（In-loop）第N遍速率控制

【0395】在一个实施例中，内环路第 N 遍速率控制使用策略和使用映射，以作出关于量化因子和其它参数对压缩帧大小和感觉失真影响的最大可能预测。采用短时校正策略可能产生偏离策略的受限判断。通常而言，遵循策略将会阻止对 VBV 模型的违反。在一个实施例中，回路第 N 遍速率控制使用了 PID 控制环路。在控制环路中的反馈是从理想比特率偏离的积累。

【0396】尽管策略产生并不包括丢弃帧，但是内环路第  $N$  遍速率控制可丢弃帧，否则 VBV 缓冲器将会下溢。同样地，回路第  $N$  遍速率控制可需要插入视频填充从而阻止 VBV 溢出。

### 3.5.6 预测

【0397】在一个实施例中，以这样的方式来执行 AD/DC 预测：该方式与参考了 ISO/IEC 14496-2: 2001(E), 7.4.3 节 (DC 和 AC 预测) 及 7.7.1 节 (区域 DC 和 AC 预测) 的标准相一致。

### 3.5.7 纹理编码

【0398】根据本发明实施例的编码器可以这样的方式来执行纹理编码：该方式与参考了 ISO/IEC 14496-2: 2001(E), 附录 B (变长编码) 及 7.4.1 节 (变长解码) 的标准相一致。

### 3.5.8 运动编码

【0399】根据本发明实施例的编码器可以这样的方式来执行运动编码：该方式与参考了 ISO/IEC 14496-2: 2001(E), 附录 B (变长编码) 及 7.6.3 节 (运动矢量解码) 的标准相一致。

### 3.5.9 产生“视频”块

【0400】视频轨可被认为是一列帧，从 1 到  $N$ 。根据本发明实施例的系统能够编码该序列以产生压缩的比特流。通过将比特流分段为块 1 到  $N$ ，来格式化比特流。每个视频帧  $n$  具有相应的块  $n$ 。

【0401】通过将来自比特流的比特附加到块  $n$  来产生块，直到块  $n$  以及块 1 至块  $n-1$  包含了足够的信息为止，该信息用于根据本发明实施例的解码器来解码视频帧  $n$ 。在块 1 直到块  $n-1$  包含了足够产生视频帧  $n$  的信息的实例中，根据本发明实施例的编码器可包括标记块。在一个实施例中，标记块是未编码的、具有和前面帧相同的定时信息的 P-帧。

## 3.6 产生“字幕”块

【0402】根据本发明实施例的编码器可在一系列标准格式之一中接受字幕，并接着将该字幕转换为位图。位图中的信息然后使用游程编码

而被压缩。该游程编码的位图被格式化到块内，其还包含了关于包含在块之内的特定字幕的开始时间和结束时间。在若干个实施例中，关于字幕在显示器上颜色、大小和位置的信息也可被包含在块中。块可被包含到字幕轨中，其为字幕设置调色板，并指示出调色板已经改变。可使用任何能够以标准字幕格式产生字幕的应用程序，来产生字幕的文本。或者，可使用软件而将由用户输入的文本直接转换为字幕信息。

### **3.7 交错**

**【0403】**当交错器收到了所有上文所描述的块之后，交错器就组建一个多媒体文件。组建多媒体文件包括创建“CSET”块、“INFO”列表块、“hdrl”块、“movi”列表块，以及idx1块。根据本发明实施例的、用于创建这些块和产生多媒体文件的方法将在下文中描述。

#### **3.7.1 产生“CSET”块**

**【0404】**如上所述，“CSET”块是可选的，并且可由交错器根据 AVI 容器格式规范来产生。

#### **3.7.2 产生“INFO”列表块**

**【0405】**如上所述，“INFO”列表块是可选的，并且可由交错器根据 AVI 容器格式规范来产生。

#### **3.7.3 产生“hdrl”列表块**

**【0406】**“hdrl”列表块由交错器基于各种提供到交错器的块中的信息来产生。“hdrl”列表块引用了所引用的块在文件之内的位置。在一个实施例中，为了建立引用，“hdrl”列表块使用文件偏移（file offset）。

#### **3.7.4 产生“movi”列表块**

**【0407】**如上所述，“movi”列表块通过编码音频、视频和字幕轨，以创建“音频”、“视频”和“字幕”轨，并接着将这些块交错而产生。在若干个实施例中，“movi”列表块还可包括数字版权管理信息。

##### **3.7.4.1 交错视频/音频/字幕**

【0408】交错音频、视频和字幕轨可使用多种规则。典型地，交错器为每个视频和音频轨建立许多队列。交错器确定哪个队列应被写入输出文件。队列选择可以基于交错周期：其从具有最少数目的所写交错周期的队列写入。在块被写入文件之前，交错器可能不得等到队列中出现整个的交错周期。

【0409】在一个实施例中，交错所产生的“音频”、“视频”和“字幕”块，以使得“音频”和“字幕”块位于文件中“视频”块之前，并且“视频”块包含了关于它们所对应的视频帧的信息。在其它实施例中，“音频”和“字幕”块可被放置在它们所对应的“视频”块之后。在“音频”、“视频”和“字幕”块的位置之间的时间差异主要依赖于用于播放这些装置的播放器的缓冲能力。在缓存受限或未知的实施例中，交错器这样交错“音频”、“视频”和“字幕”块：“音频”和“字幕”块位于“视频”块之间，其中所述“视频”块紧跟着“音频”和“字幕”块，其包含与音频或字幕相对应的第一视频帧。

#### 3.7.4.2 产生DRM信息

【0410】在DRM被用来保护多媒体文件的视频内容的实施例中，DRM信息可在视频块编码的同时产生。在产生每个块时，该块可被加密并且所产生的DRM块包含了关于视频块加密的信息。

#### 3.7.4.3 交错DRM信息

【0411】根据本发明实施例的交错器将DRM块交错在视频块之前，所述DRM块包含了关于视频块加密的信息。在一个实施例中，视频块 $n$ 的DRM块位于视频块 $n-1$ 和视频块 $n$ 之间。在其它实施例中，DRM在视频块 $n$ 之前和之后的距离依赖于配备在解码多媒体文件的装置之内的缓存的数量。

#### 3.7.5 产生“idxl”块

【0412】一旦“movi”列表块已经被产生，“idxl”块的产生是一个简单的过程。通过读取“movi”列表块之内的每个“数据”块的位置，可创建“idxl”块。这一信息与读取自“数据”块的关于“数据”块属于哪个轨以及“数据”块的内容的信息相结合。所有这些信息然后以



这样的方式被插入到“idx1”块中：对于用来表现信息的上文描述的格式，该方式是适当的。

#### 4. 多媒体文件的传输和分发

【0413】一旦产生了多媒体文件，就可经由任意多个网络分发该文件。如下事实简化了信息的传输：在许多实施例中，产生多媒体影像和菜单所需的在其它事物之间的元素，被包含在单一文件中。在若干个实施例中，多媒体文件可与解密多媒体文件内容所需的信息分离地分发。

【0414】在一个实施例中，多媒体内容被提供给第一服务器，并被编码以创建根据本发明实施例的多媒体文件。然后，该多媒体文件可被放置在第一服务器或第二服务器。在其它实施例中，DRM 信息可位于第一服务器、第二服务器或第三服务器。在一个实施例中，可查询第一服务器以确认编码的多媒体文件的位置和/或确认 DRM 信息的位置。

#### 5. 解码多媒体文件

【0415】可通过以下装置访问来自本发明实施例的多媒体文件的信息：被配置成使用合适软件的计算机、专用播放器（其是访问多媒体文件信息的硬件），或任何其它能够解析 AVI 文件的装置。在多个实施例中，装置可访问多媒体文件中的所有信息。在其它实施例中，装置可能不能访问根据本发明实施例的多媒体文件中的所有信息。在一个特定实施例中，装置不能访问上述存储在非 AVI 文件格式的块中的任何信息。在不能访问所有信息的实施例中，该装置通常将会丢弃那些不能被该装置识别的块。

【0416】典型地，有能力访问包含在根据本发明实施例的多媒体文件中的信息的装置可以执行许多功能。该装置可显示多媒体影像，其包括在可视化显示器上显示视频，从潜在地处于音频系统的多个音频轨之一产生音频，以及显示来自潜在地许多字幕轨之一的字幕。几个实施例在伴随着音频和/或视频播放的同时，还可在可视化显示器上显示菜单。这些显示菜单是交互式的，具有功能部件例如可选按钮、下拉菜单和子菜单。在一些实施例中，菜单项可指向正被访问的多媒体文件外部的音频/视频内容。外部内容可位于访问多媒体文件的装置本地，

或被远程地放置，例如通过局域网、广域网或公网。许多实施例还可搜索一个或多个多媒体文件，其根据包含在多媒体文件（一个或多个）内的“元数据”，或由一个或多个多媒体文件所引用的“元数据”。

### **5.1 多媒体影像的显示**

**【0417】** 本发明实施例的多媒体文件被给予了支持多个视频轨、多个音频轨和多个字幕轨的能力，对多媒体影像的显示使用了组合了视频、音频和/或字幕的多媒体文件，其可经由可视化菜单系统或下拉菜单系统（下面将讨论其操作）或通过用于产生多媒体影像的装置的默认设置，来强制选择特定的音频轨、视频轨和/或字幕轨。当音频轨、视频轨和潜在地字幕轨被选择时，多媒体影像的显示可以继续。

**【0418】** 在图 4.0 中示出了根据本发明实施例的处理过程，其用于定位所需的、来自包含 DRM 的多媒体文件的多媒体信息以及用于显示该多媒体信息。处理过程 620 包括获得解密 DRM 头所需的加密密钥 622。接着，使用该加密密钥来解密（624）DRM 头，并将第一 DRM 块布置在多媒体文件的“movi”列表块之内。从“DRM”头中的表获得（628）解密“DRM”块所需的加密密钥，并使用该密钥以解密加密的视频块。接着，伴随视频块所需的音频块和任何所需的字幕块被解码（630），并且该音频、视频和字幕信息通过显示器和音响系统而被表现（632）。

**【0419】** 在若干个实施例中，所选的音频轨包括多个通道或声道，以提供立体声和环绕声音频。当字幕轨被选择而被显示时，可对前面的视频帧是否包含字幕作出判决（这一判决可以多种方式中的任何一种来作出，其可获得这样的成果：标识出前面包含字幕信息的“字幕”块，该信息将通过正在被解码的视频帧显示）。如果前面的字幕包括了用于字幕的、指示出字幕应由当前帧来显示的字幕信息和时间选择信息，则字幕就被叠加在解码的视频帧上。如果前面的帧没有包括用于前面帧上字幕的、指示出字幕不应联合当前解码帧来显示的字幕信息和时间选择信息，则寻找所选字幕轨的字幕块。如果“字幕”块被定位，则该字幕就被叠加在所解码的视频上。然后，该具有伴随音频的视频（包括任何叠加的字幕）就被显示。

【0420】返回图 4.0 的讨论，处理过程确定 (634) 是否有任何额外的 DRM 块。如果有，则下一 DRM 块将被定位 (626)，并且处理过程继续直到没有额外的 DRM 保留。此时，音频、视频和/或字幕轨的表现完成。

【0421】在若干个实施例中，装置可寻找到多媒体信息中某一具体的部分（例如，具有具体伴随音频轨以及可选地具有具体字幕轨的电影的具体场景），其使用了包含在根据本发明的多媒体文件的“hdl”块之内的信息。在许多实施例中，“视频”块、“音频”块和/或“字幕”块的解码可和其它任务一起并行执行。

【0422】能够从多媒体文件访问信息并联合具体音频轨和/或具体字幕轨来显示视频的装置的实例是计算机，其使用软件并以上文所述方式配置。另一个实例是 DVD 播放器，其装备了具有这些能力的编解码器。在其它实施例中，任何配置成定位或选择（不论是无意地或随意地）对应于具体媒体轨的“数据”块，并解码这些轨进行显现的装置，都可以使用根据本发明实施例的多媒体文件，来产生多媒体影像。

【0423】在若干个实施例中，装置可播放来自外部文件的多媒体信息，以及来自多媒体文件的多媒体信息。典型地，通过从上文所描述的类型的多媒体文件中所引用的本地文件获得 (source) 的音频轨或字幕轨，这一装置将做到这些。如果所引用的文件没有在本地存储，该装置可通过网络到达该装置所存储的位置，接下来该装置可获得文件的本地拷贝。然后，该装置可访问所有文件，建立视频、音频和字幕（如果需要的话）传递途径，通过该传递途径可从不同的文件源供给多媒体体的不同轨。

## 5.2 菜单的产生

【0424】在图 4.1 中示出了根据本发明实施例的解码器。解码器 650 通过将文件提供到多路分配器 654，而处理根据本发明实施例的多媒体文件 652。该多路分配器从多媒体文件提取“DMNU”块，并从“DMNU”块提取所有的“LanguageMenus”块，接着将它们提供给菜单语法分析器 656。多路分配器还从“DMNU”块提取所有的“Media”块，并将

它们提供给媒体渲染器 658。菜单语法分析器 656 对来自“LanguageMenus”块的信息进行语法分析，以建立表现定义于“LanguageMenus”块中的菜单结构的状态机。表现菜单结构的状态机可被用来提供到用户的显示以及响应用户命令。状态机被提供到菜单状态控制器 660。该菜单状态控制器保持对菜单状态机的当前状态地跟踪，并接收来自用户的命令。来自用户的命令可以造成状态转移。提供给用户的初始显示以及对伴随着菜单状态转移的显示的任何更新可使用菜单播放器界面或接口 662 来控制。菜单播放器界面 662 能够被连接到菜单状态控制器和媒体渲染器。该菜单播放器界面指导媒体渲染器哪些媒体应当从媒体块中被提取并通过连接到媒体渲染器的播放器 664 提供给用户。用户可使用诸如键盘、鼠标或远程控制的输入装置，向播放器发出指令。多媒体文件一般指示将菜单初始显示给用户，然后用户指令指示在初始菜单产生之后显示的音频和视频。图 4.1 所示的系统可使用计算机和软件来实现。在其它实施例中，可使用功能特定集成电路或软件和固件的结合来实现系统。

**【0425】**在图 4.2 中示出了根据本发明实施例的菜单的实例。菜单显示器 670 包括四个按钮区域 672，包括标题 676 和指示器 678 的背景视频 674。菜单还包括背景音频（未示出）。显示器所创建的可视化效果可以是具有欺骗性的。按钮的外观有代表性地作为背景视频的一部分，并且按钮自身是背景视频的限定区域，在该区域被指示器所激活时，其具有与这些区域相关联的特别的动作。指示器有代表性地是覆盖物。

**【0426】**图 4.3 概念性地示出了示于图 4.2 的显示器中的所有信息的源。背景视频 674 可包括菜单标题、按钮的外观，以及显示器的背景。所有这些元素和额外的元素可以是静态或动画显现的。使用包含在“MediaTrack”块 700 中的信息可提取背景视频，所述的包含在“MediaTrack”块 700 中的信息指示了背景视频在视频轨 702 之内的位置。可伴随菜单的背景音频 706 可使用“MediaTrack”块 708 而被定位，所述“MediaTrack”块 708 指示了背景音频在音频轨 710 之内的位置。如上所述，指示器 678 是覆盖物 713 的一部分。覆盖物 713 还能够包含图形，其出现以高亮背景视频中作为按钮出现的部分。在一个实施

例中，覆盖物 713 使用“MediaTrack”块 712 获得，该“MediaTrack”块 712 指示了覆盖物在覆盖轨 714 之内的位置。菜单与用户之间交互作用的方式由与每个按钮相关联的“Action”块（未示出）来定义。在示例性实施例中，“PlayAction”块 716 是示例性的。“PlayAction”块间接地引用（其它由“PlayAction”块所引用的块未示出）包含在多媒体文件（即，音频、视频以及可能是字幕轨）之内的多媒体影像的一场景。“PlayAction”块 716 最终使用“MediaTrack”块 718 引用该场景，“MediaTrack”块 718 指示了功能部件轨之内的场景。所选的或默认的音频轨中的一点以及潜在字幕轨也可被引用。

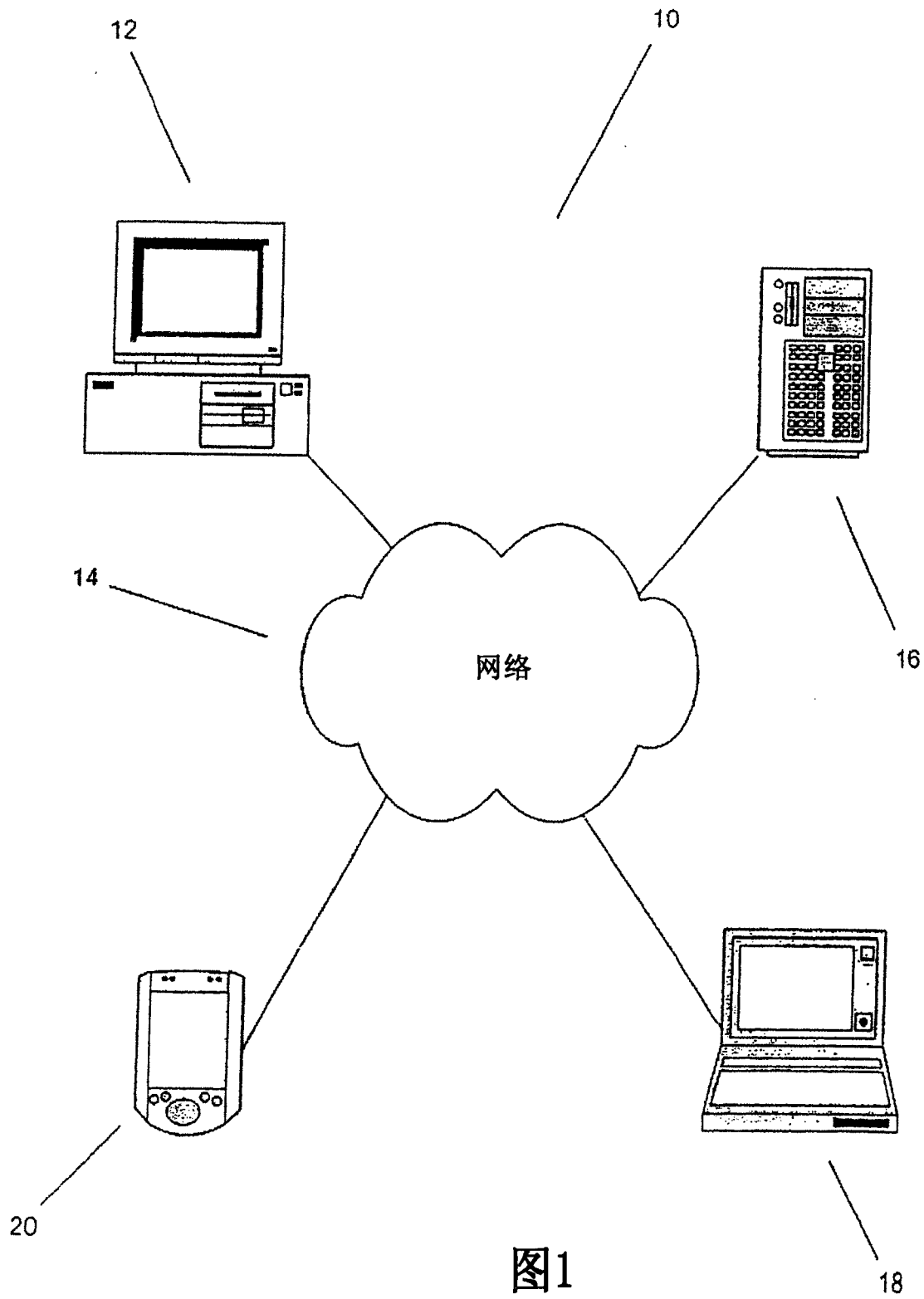
**【0427】**当用户使用输入装置输入命令时，显示器可能不仅响应于按钮区域的选择而被更新，而且可能仅仅由于指示器被定位在按钮区域之内就被更新。如上所述，用来产生菜单的所有媒体信息有代表性地被放置在多媒体文件之内，更具体的说放置在“DMNU”块之内。尽管在其它实施例中，所述信息可被放置在文件的其它位置和/或其它文件中。

### **5.3 访问元数据**

**【0428】**“元数据”是表现信息的标准化方法。“元数据”的标准化特性使得数据能够通过自动化的处理过程而被访问和理解。在一个实施例中，“元数据”可被提取并提供给用户观看。几个实施例使得在服务器上的多媒体文件可被检查，以提供关于用户观看习惯和观看喜好的信息。这种信息将被软件应用程序所使用，以推荐用户可能会喜爱观看的其它多媒体文件。在一个实施例中，推荐可基于包含在其它用户的服务器上的多媒体文件。在其它实施例中，用户能够请求一多媒体文件，而且通过搜索引擎和/或智能代理（它们将检查各个位置地多媒体文件的“元数据”）能够定位该文件。另外，用户可基于关于影像每个不同版本的编码方式的“元数据”，在包含具体多媒体影像的各种多媒体文件之间作出选择。

**【0429】**在若干个实施例中，可出于编目或创建访问文件内容的简单菜单的目的访问根据本发明实施例的多媒体文件的“元数据”。

**【0430】**虽然上面的描述包含了许多本发明的具体实施例，但这些实施例不应被解释为对本发明范围的限定，相反只是作为本发明实施例的实例而已。例如，根据本发明实施例的多媒体文件可包括单一多媒体影像或多个多媒体影像。另外，这种文件能够包含一个或多个菜单，以及任意多种的不同类型的“元数据”。因此，本发明的范围不应由所示的实施例确定，而应由所附权利要求及其等同物确定。









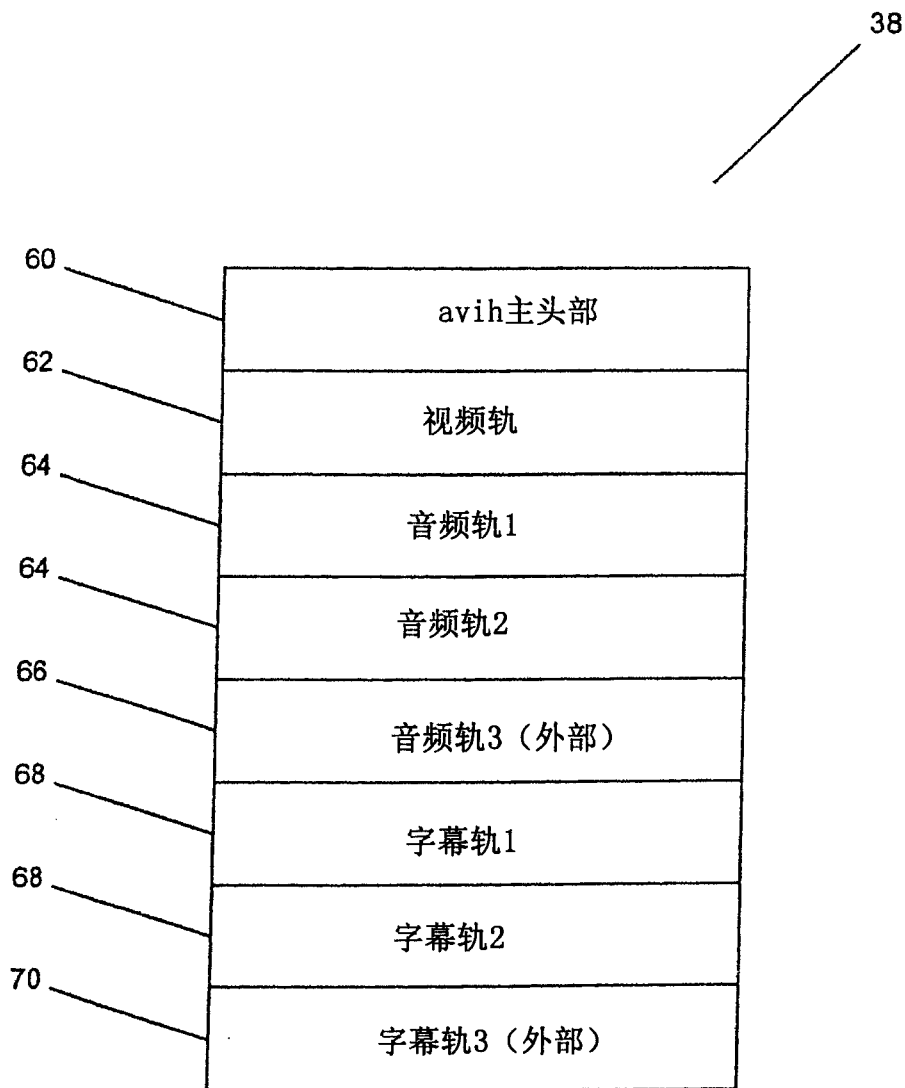


图2.1

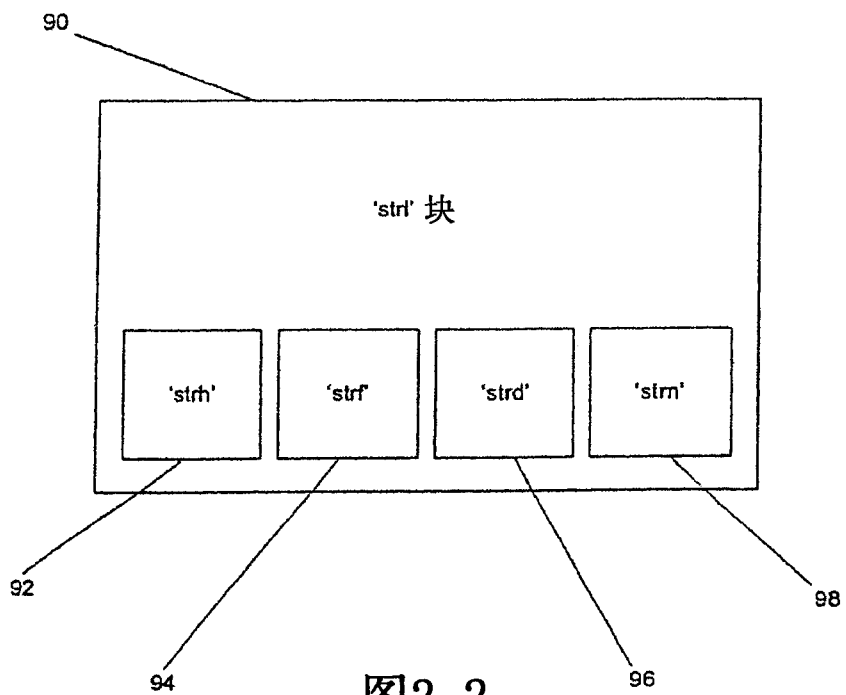


图2.2

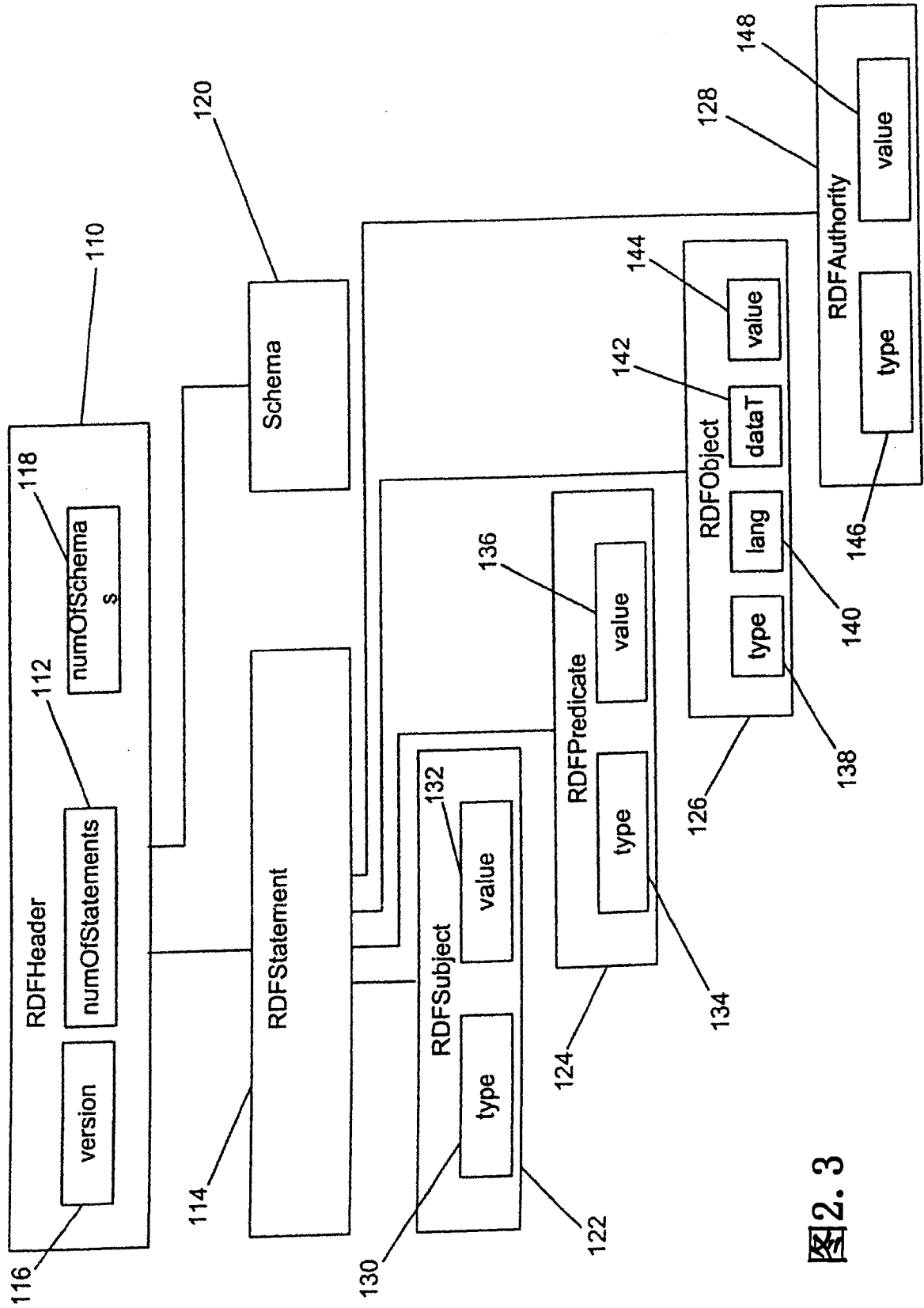


图2.3

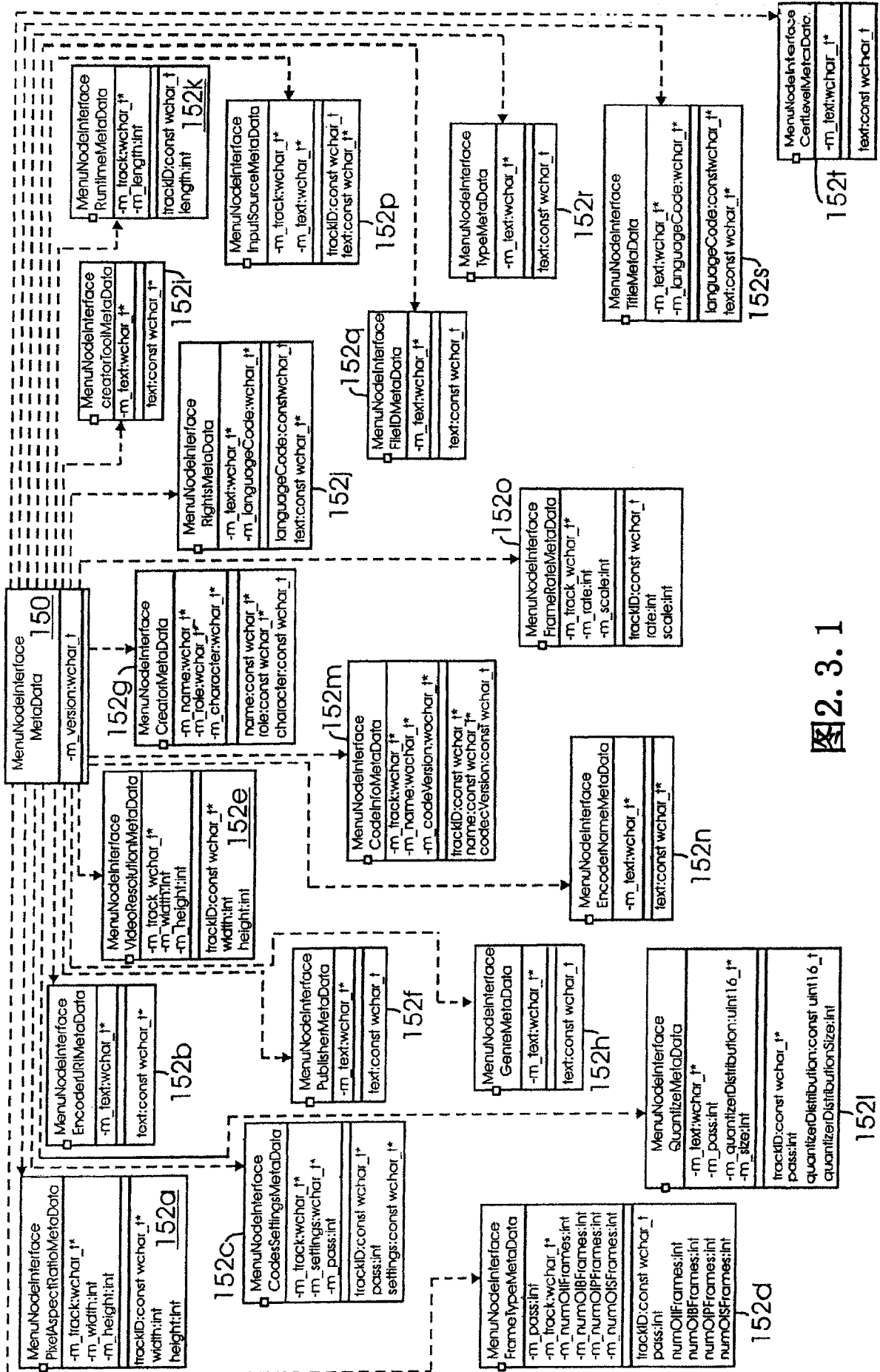
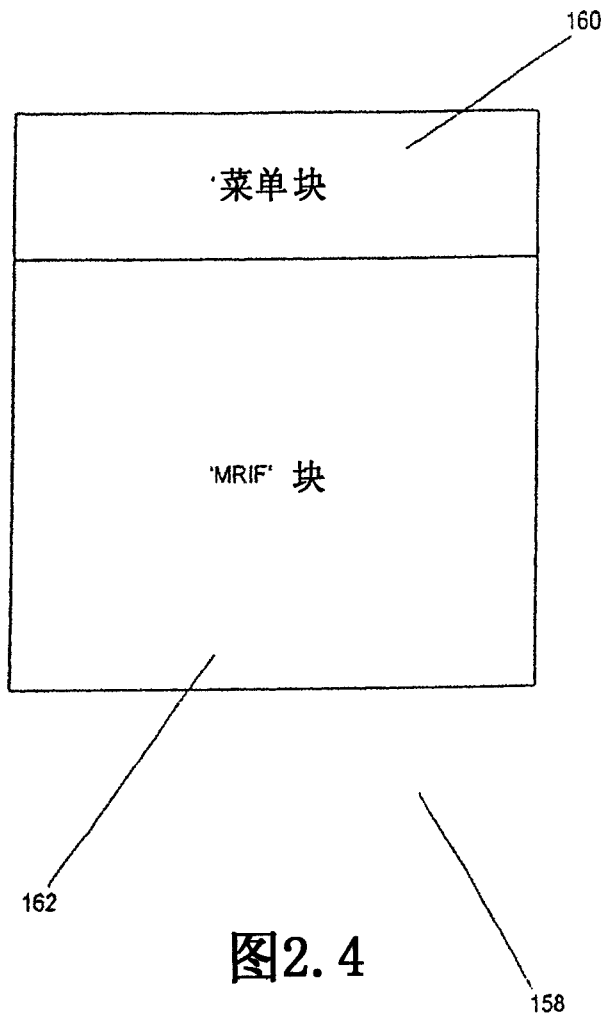


图2.3.1



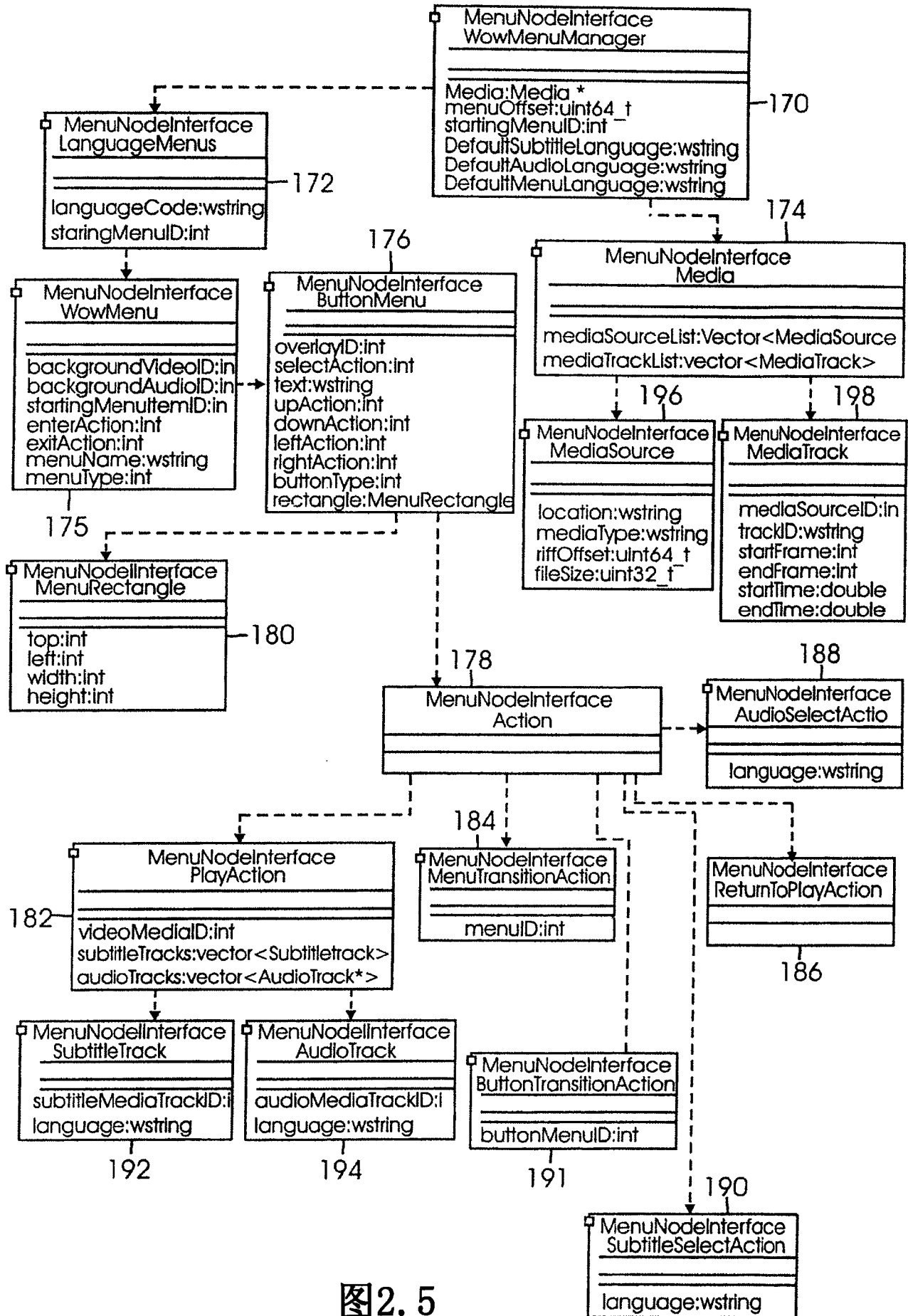


图2.5

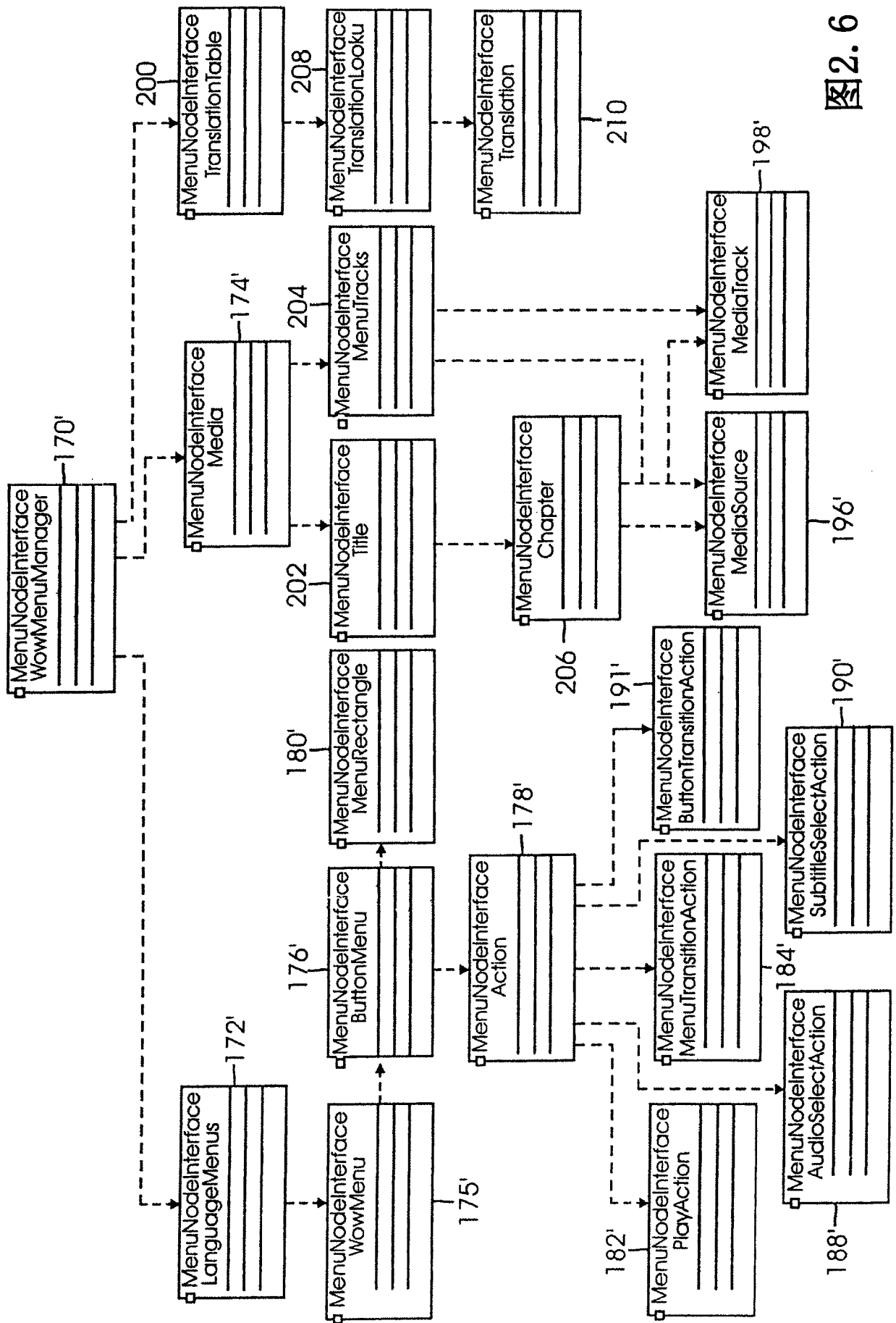


图 2.6



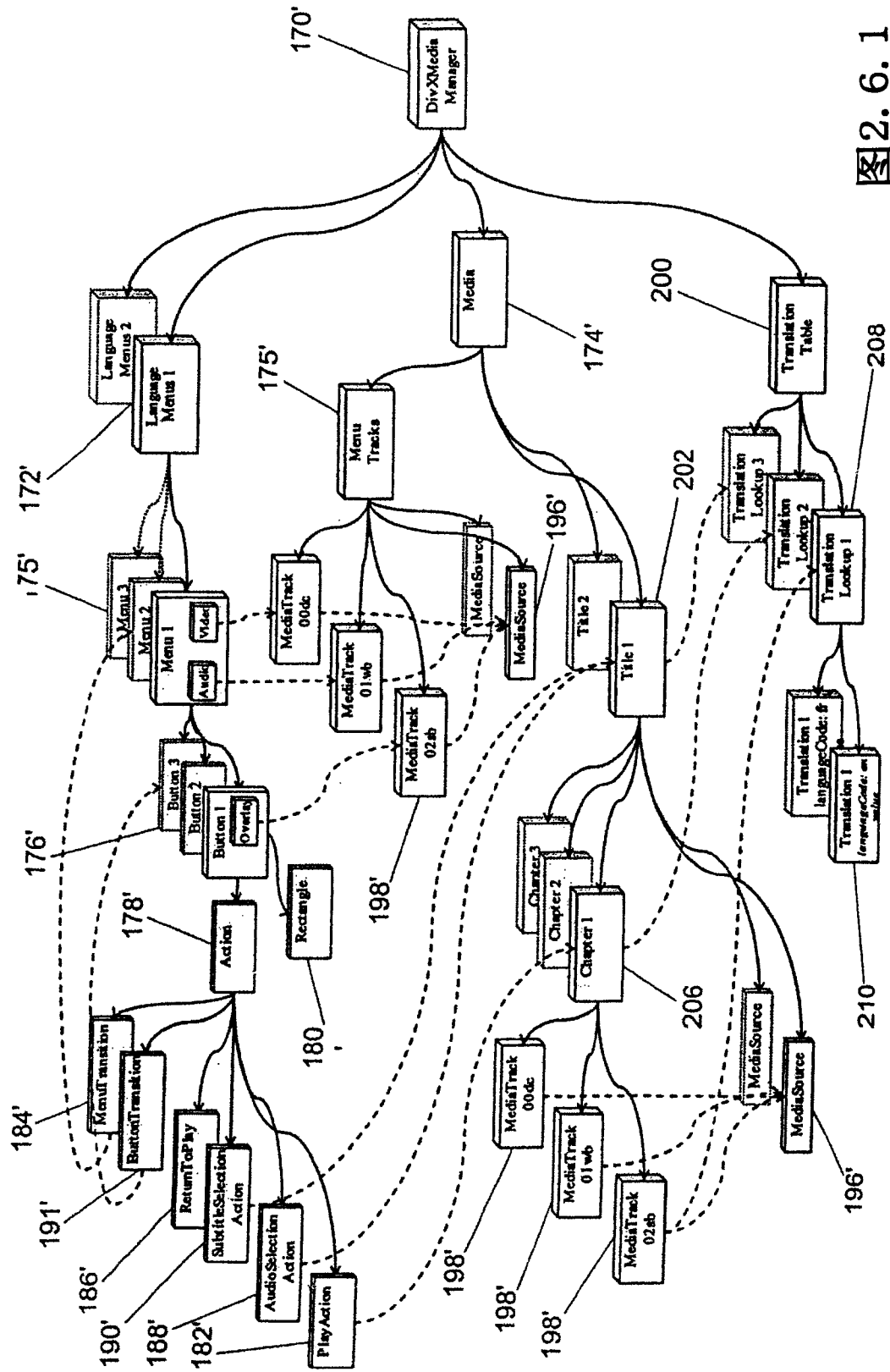


图 2.6.1

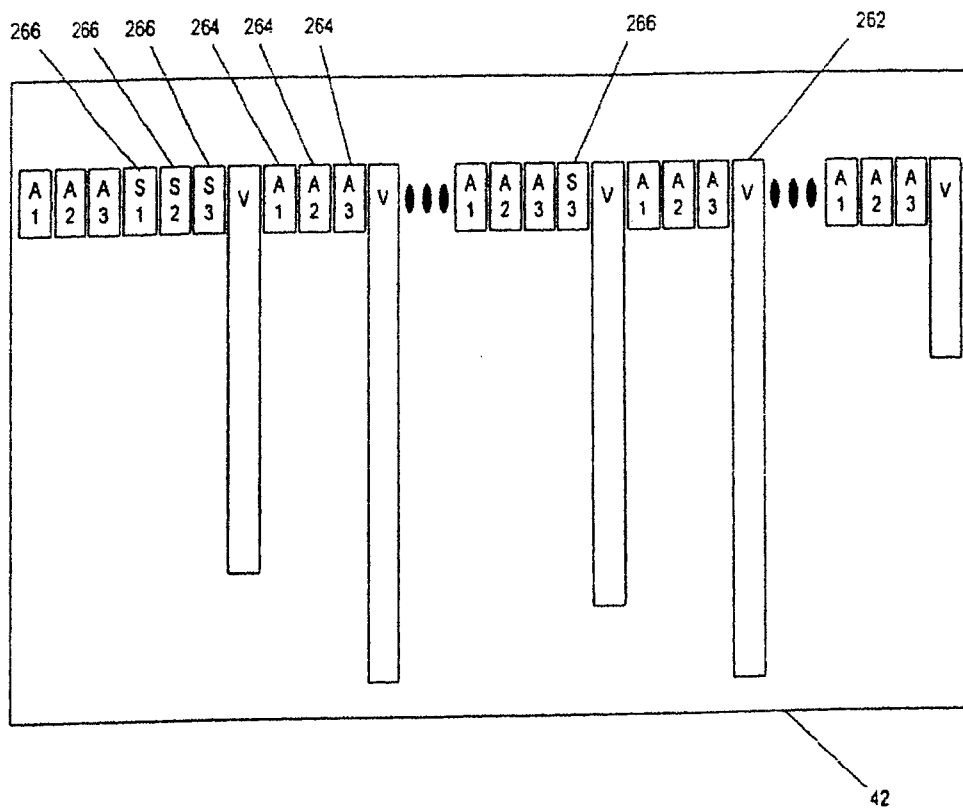


图2.7

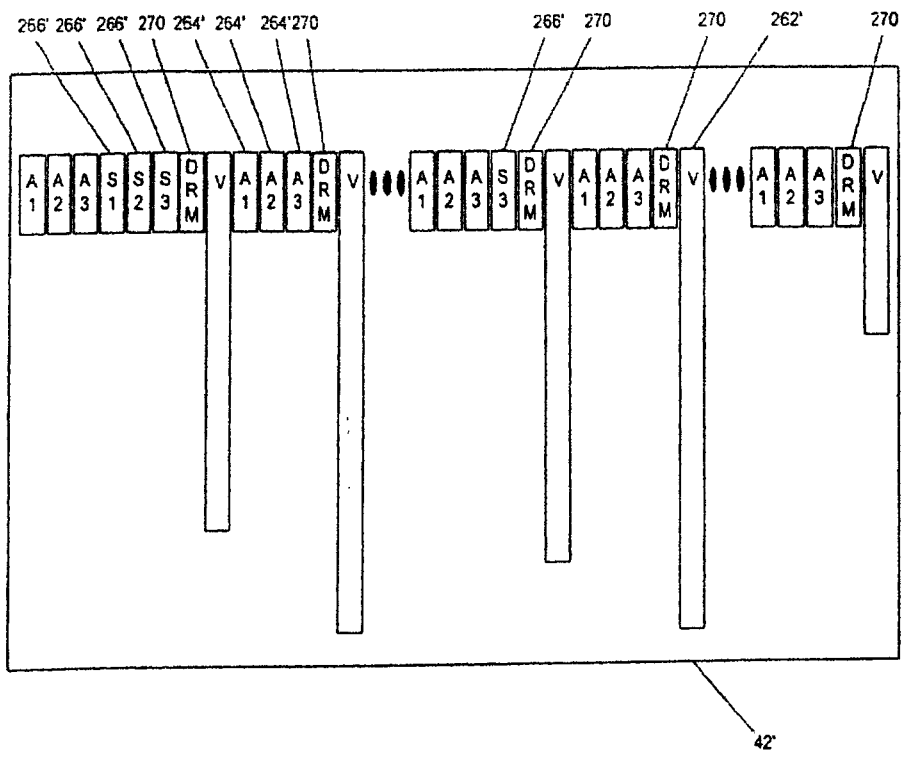


图2.8

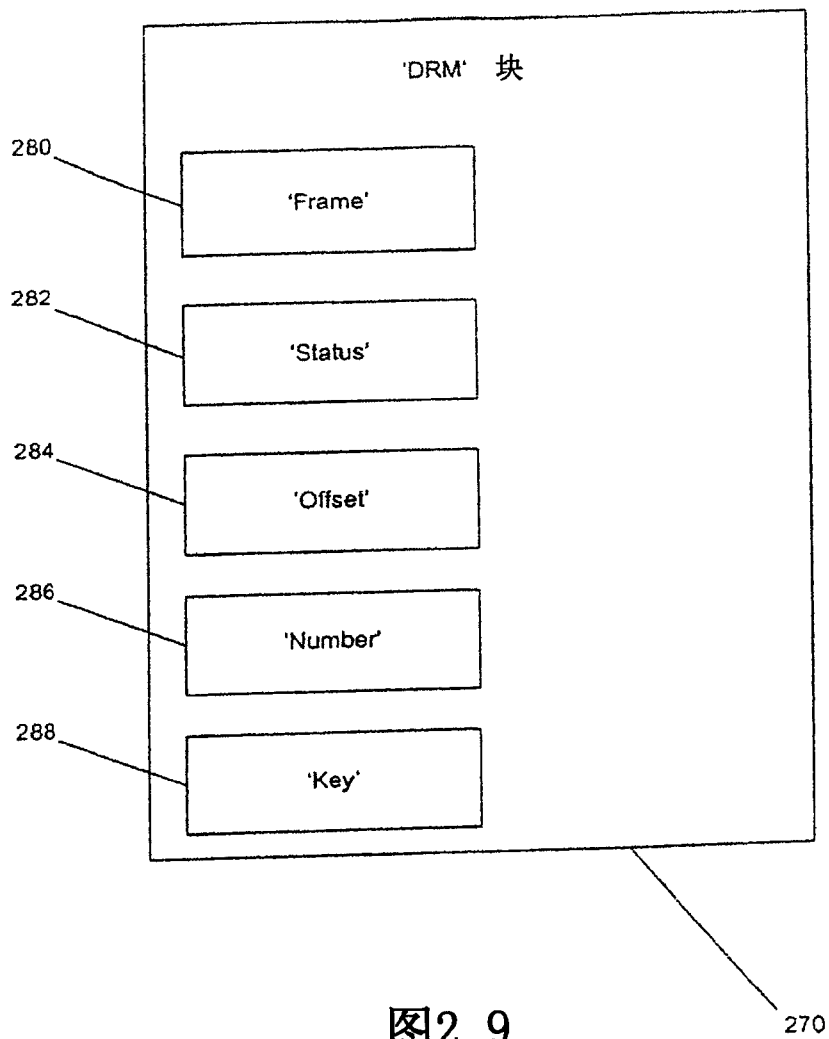


图2.9

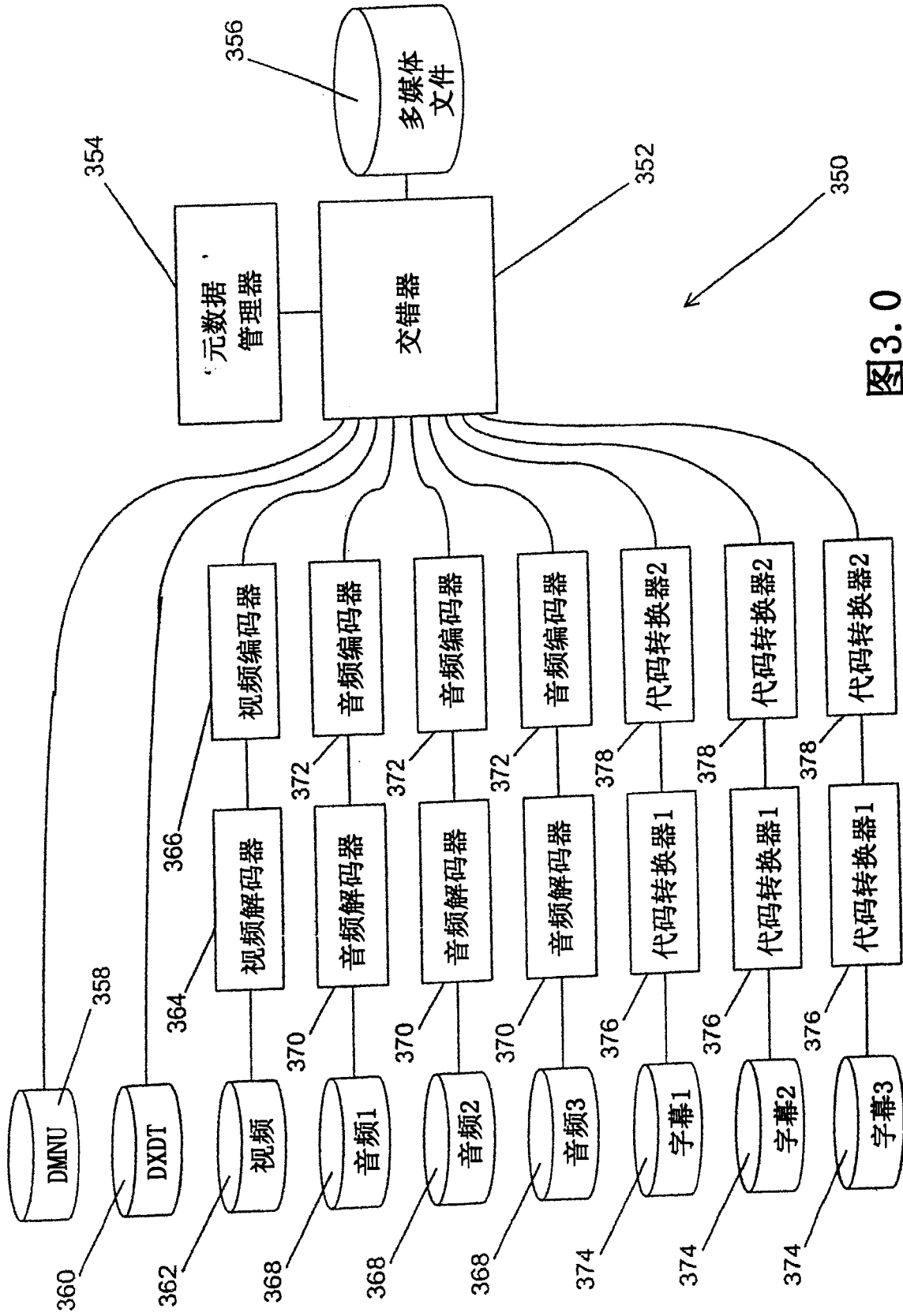


图3.0

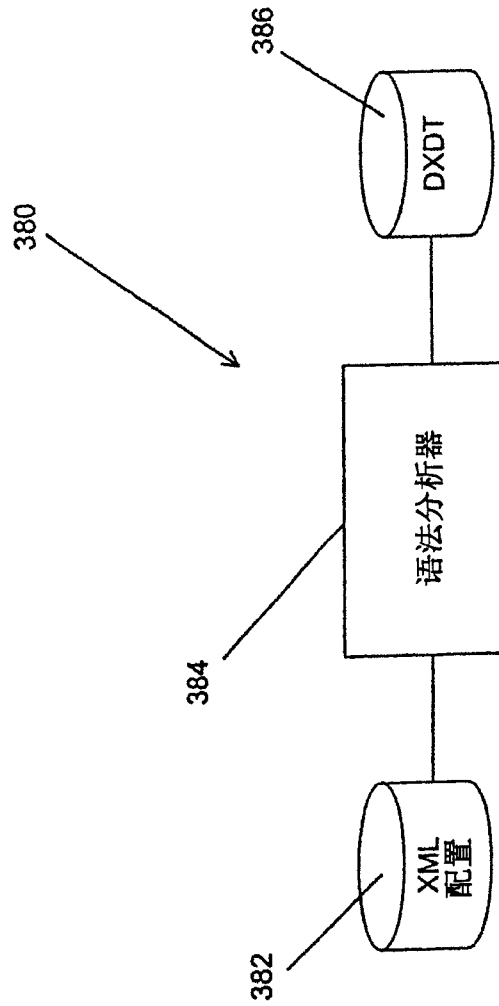


图3.1

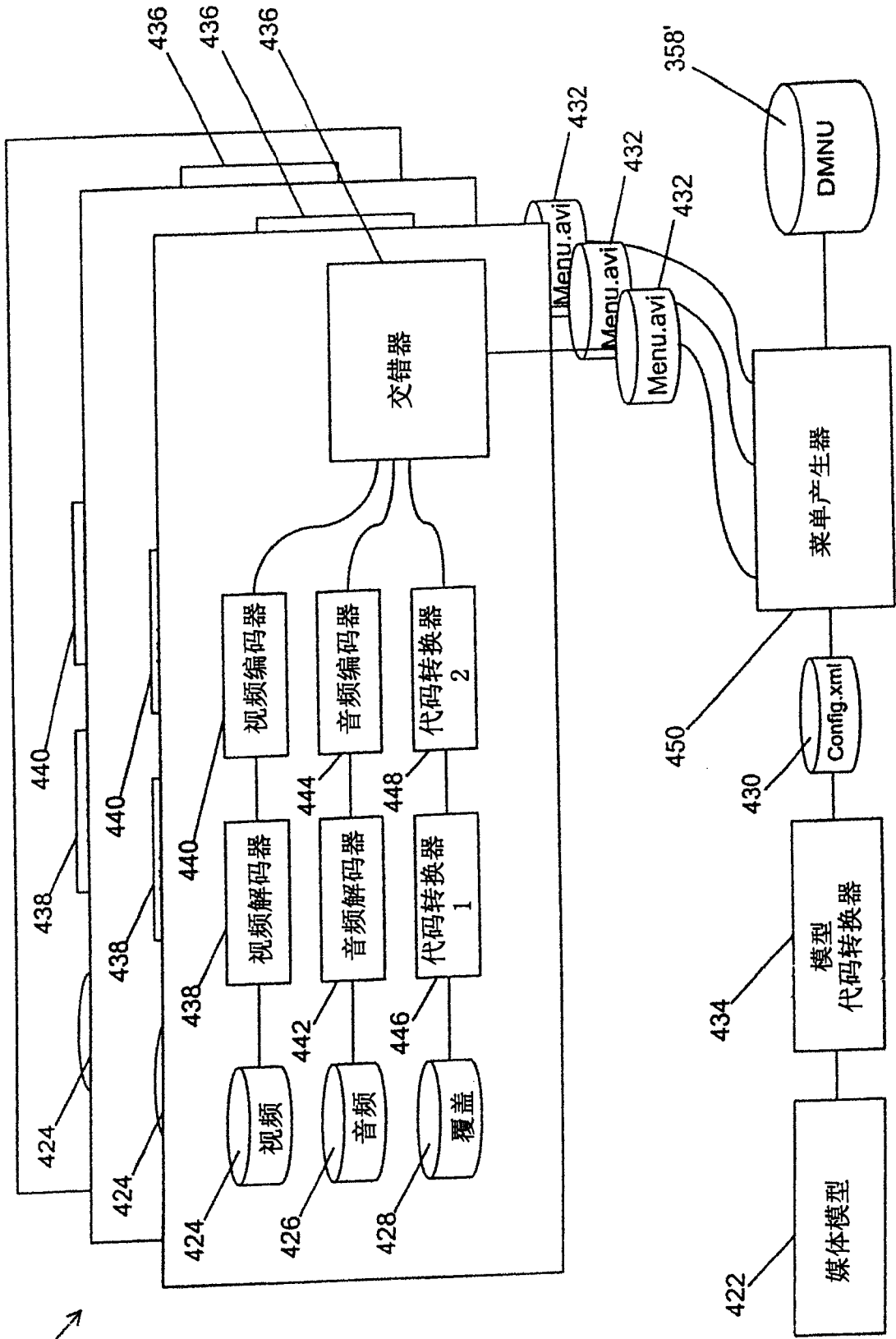


图3.2

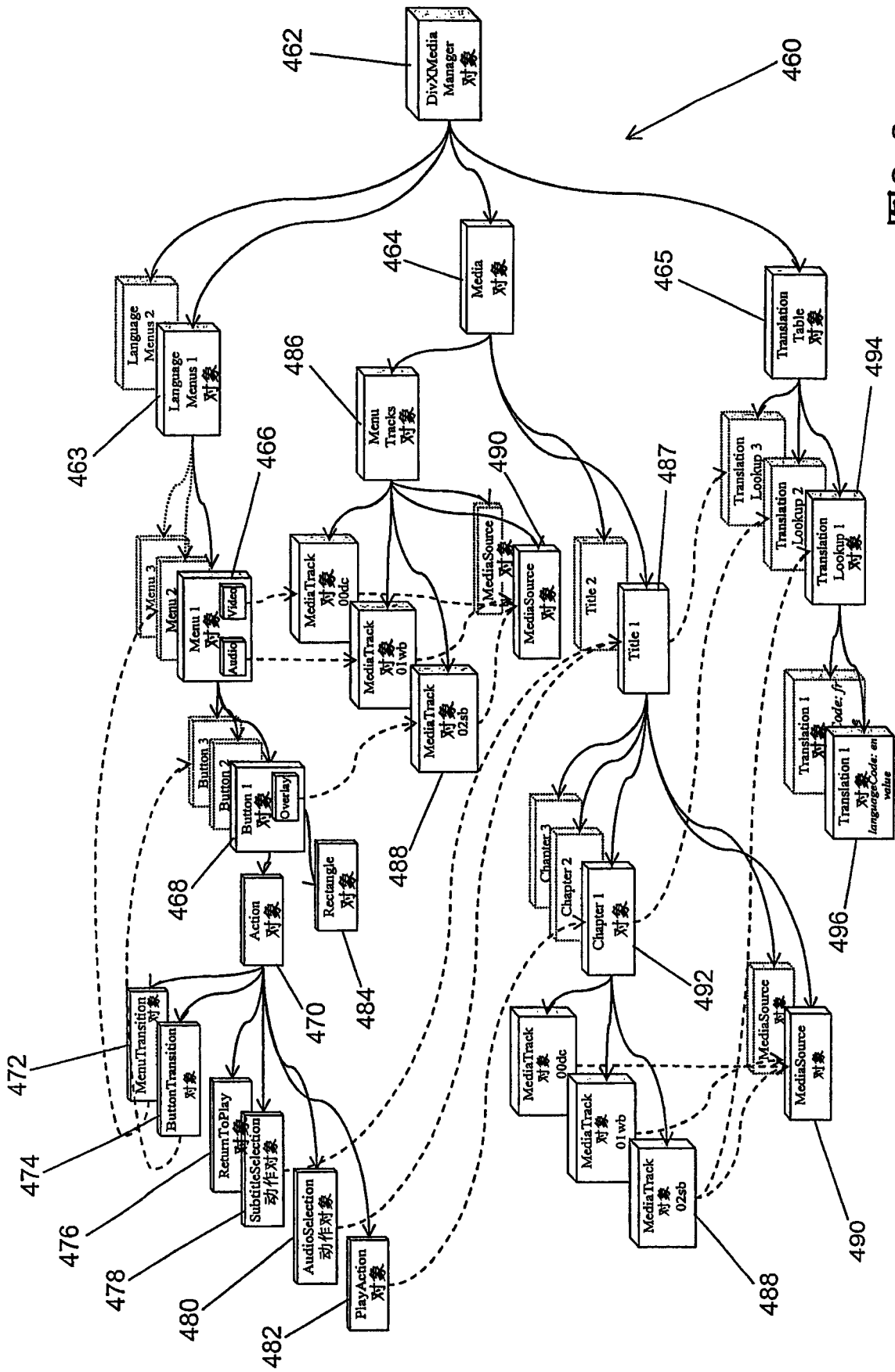


图3.3



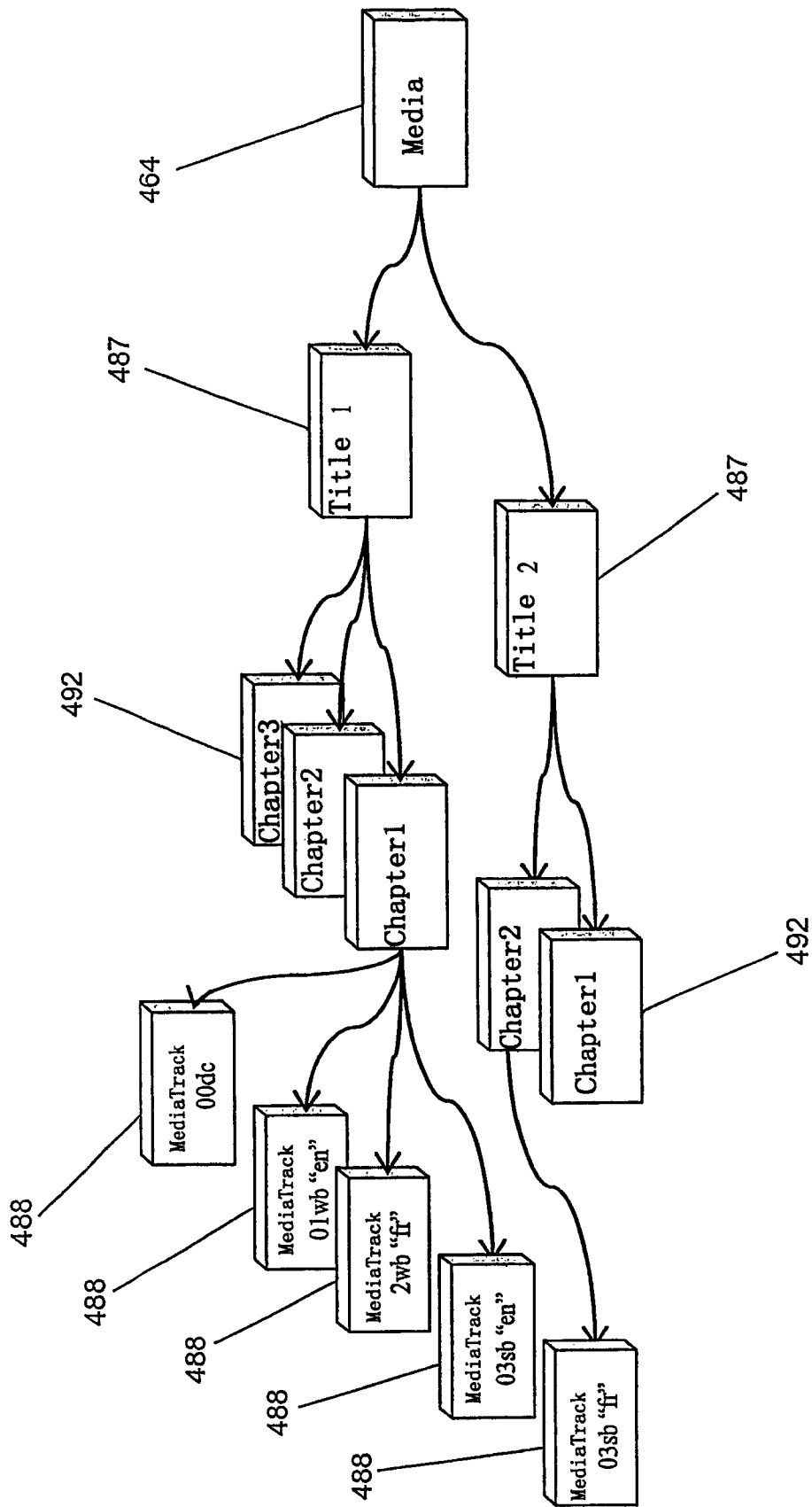


图3.3.1

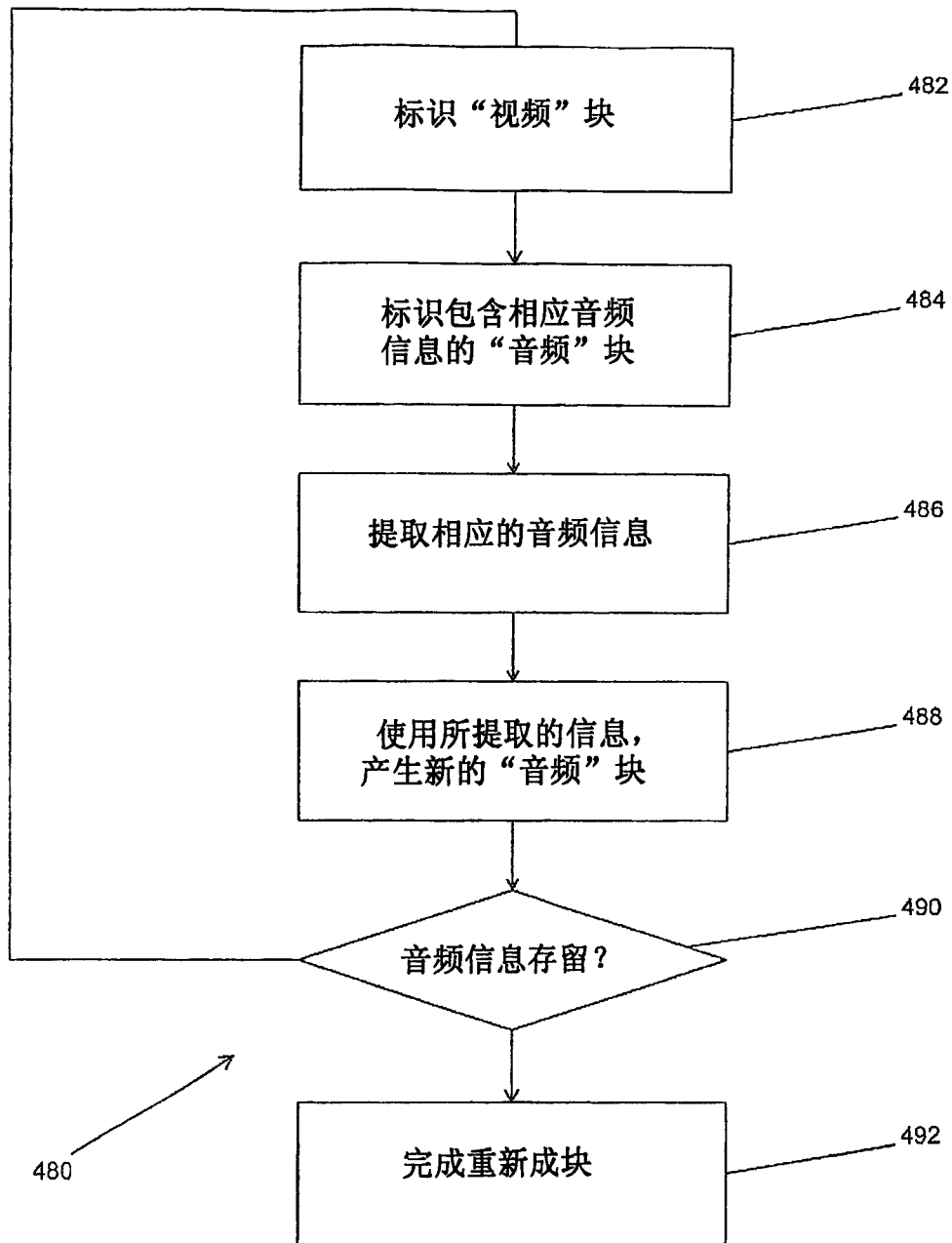


图3.4

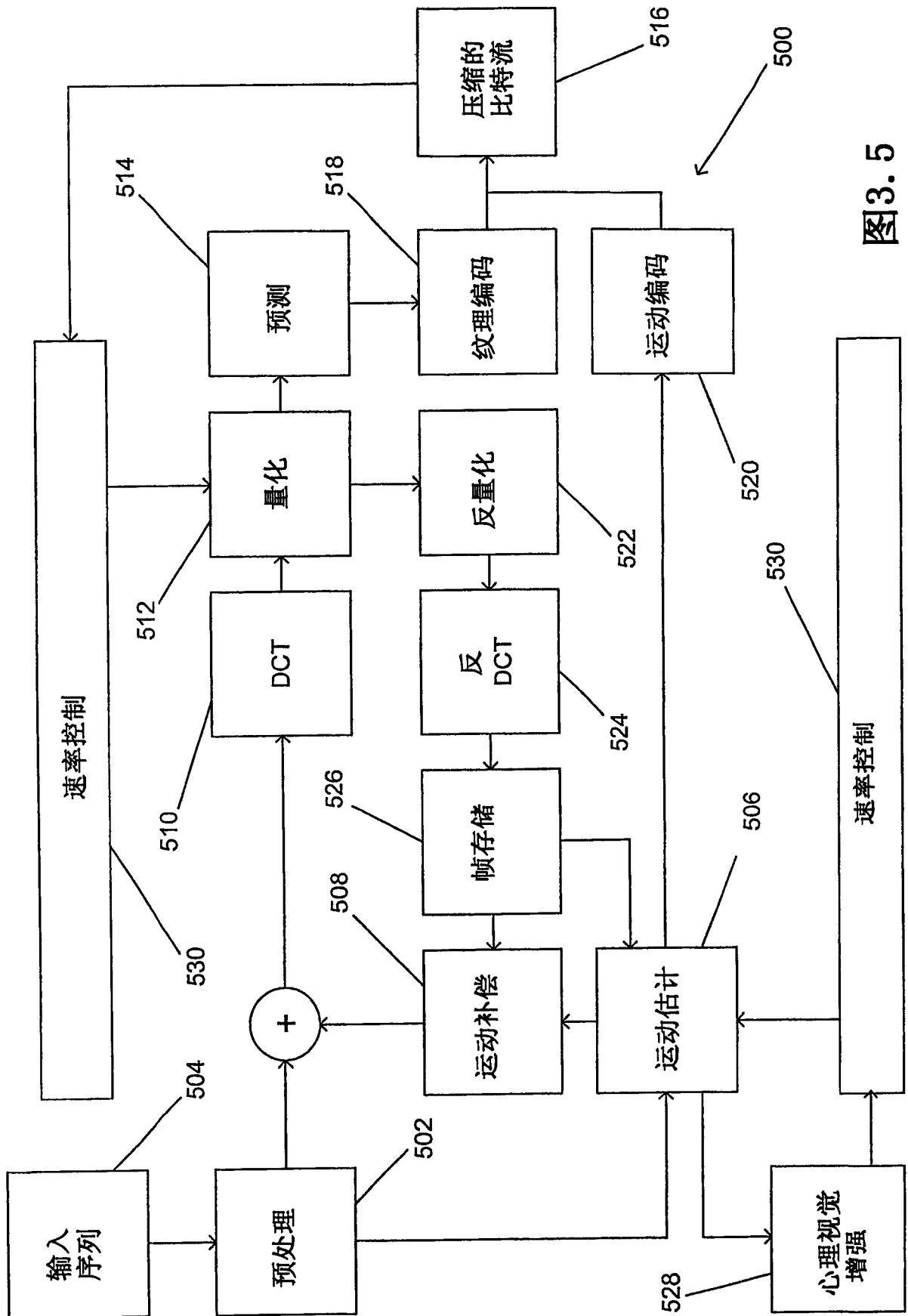


图3.5

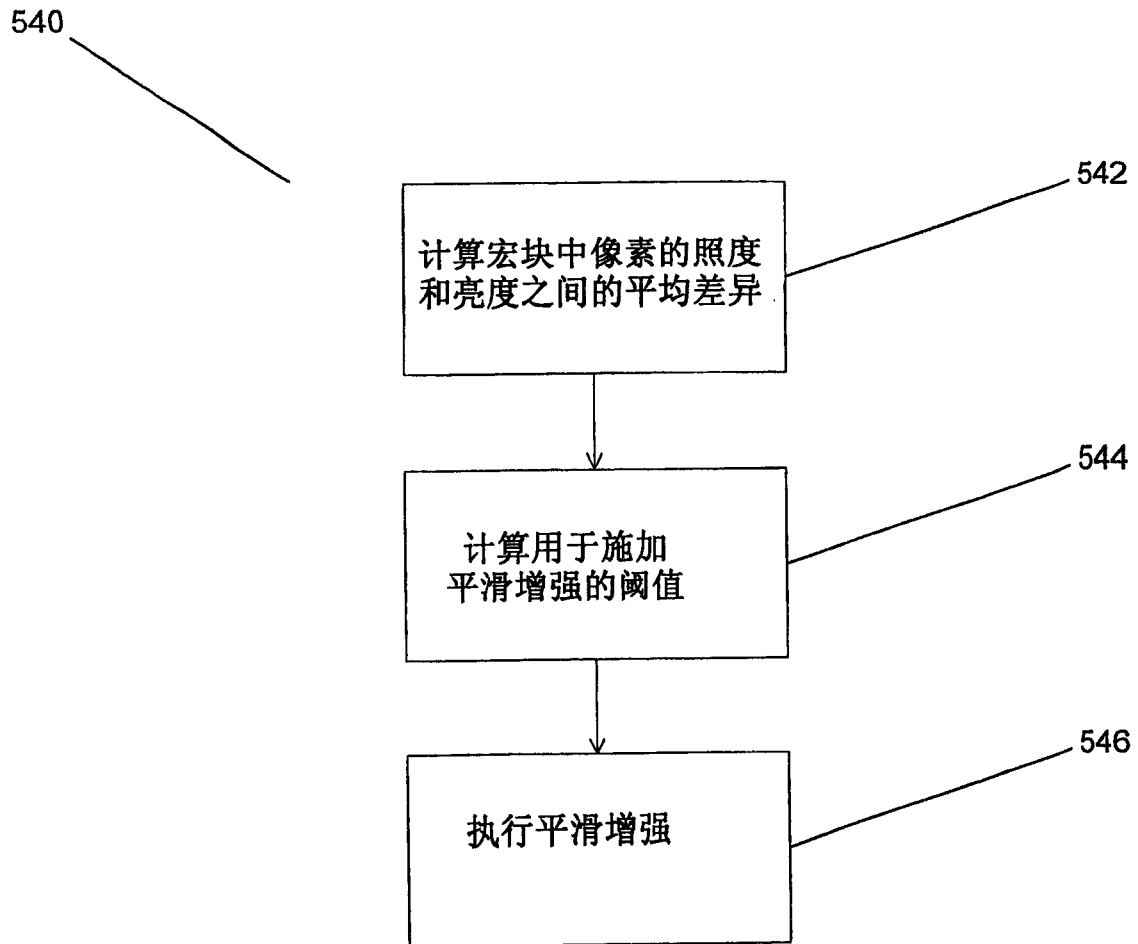


图3.6

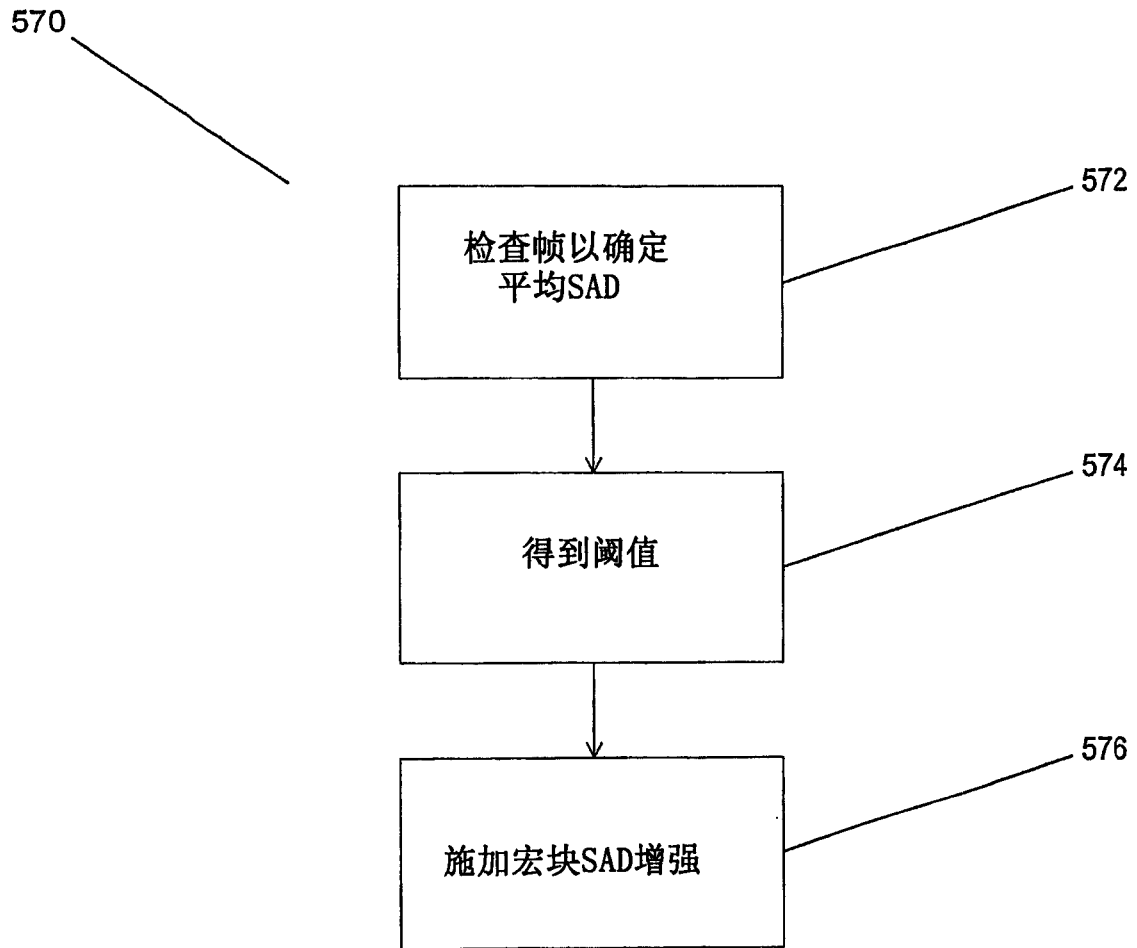


图3.7

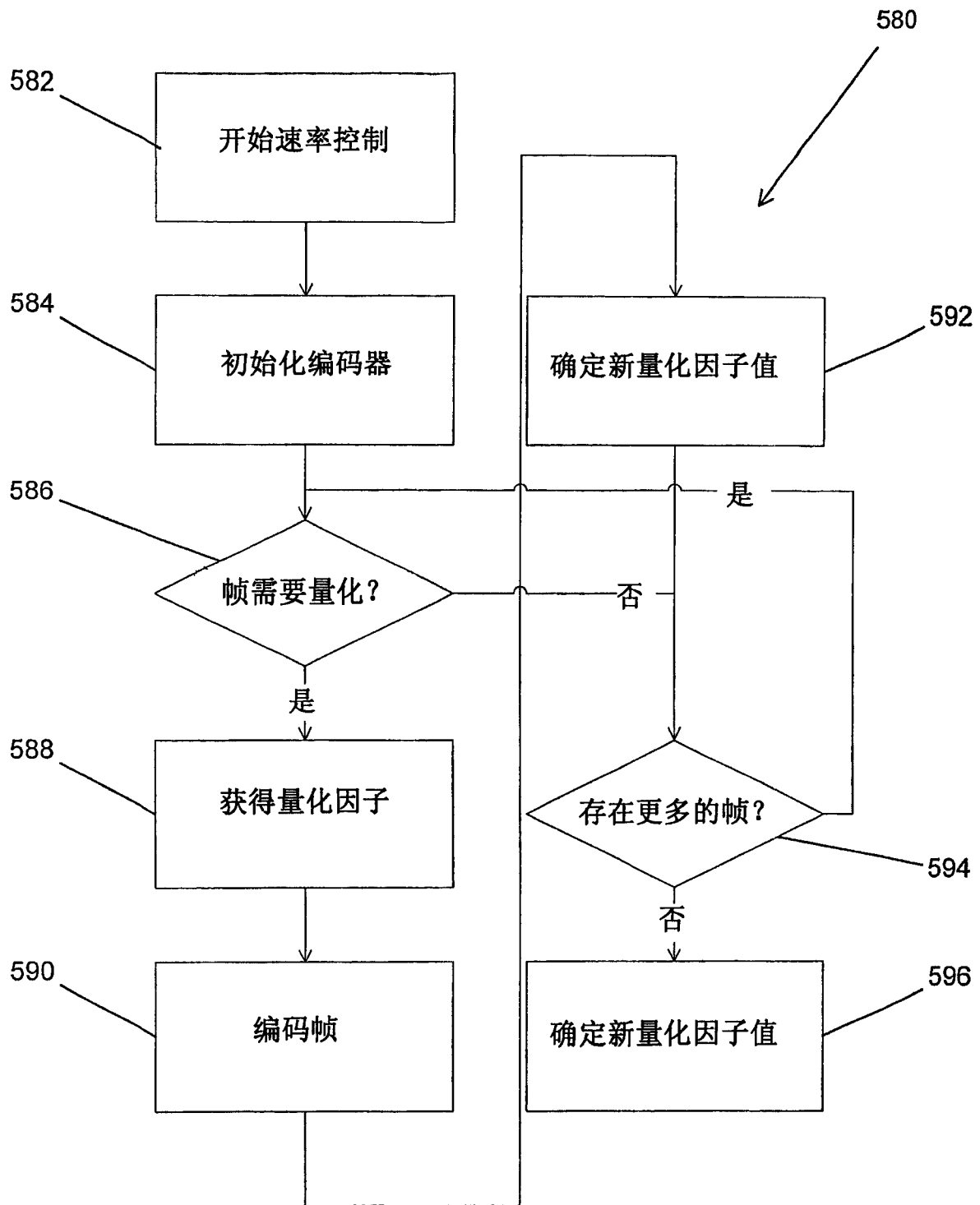


图3.8

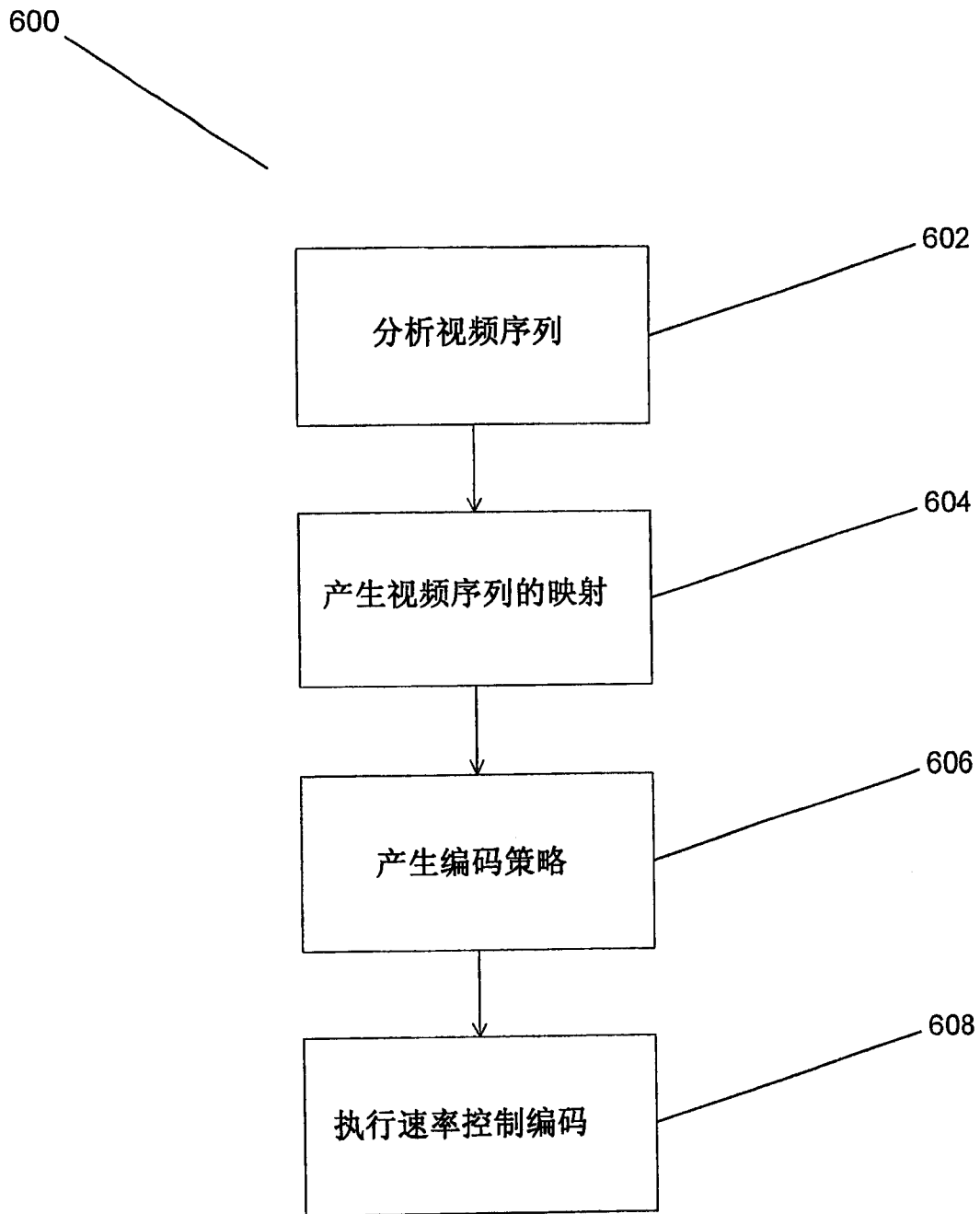


图3.9

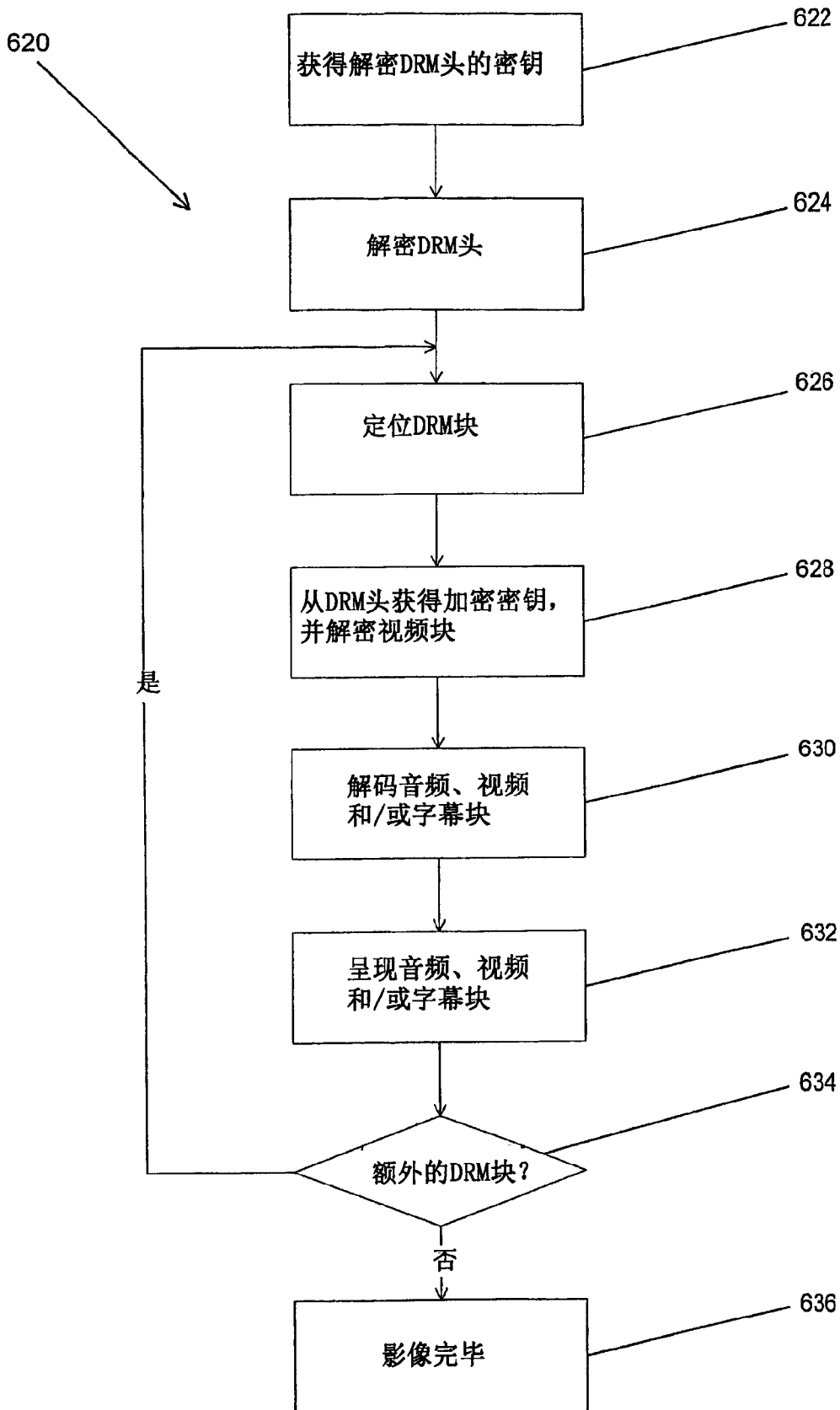


图4.0



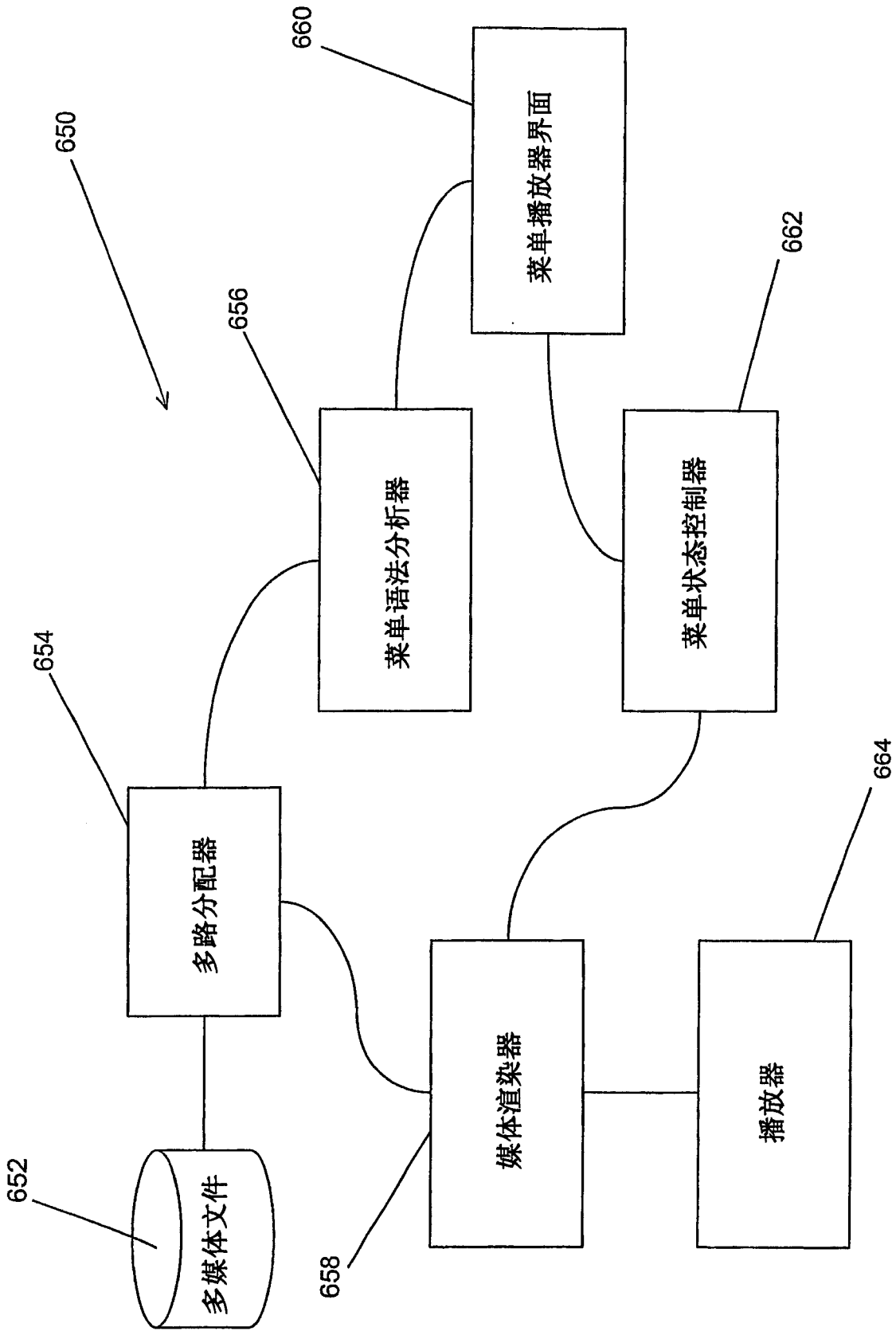


图4.1

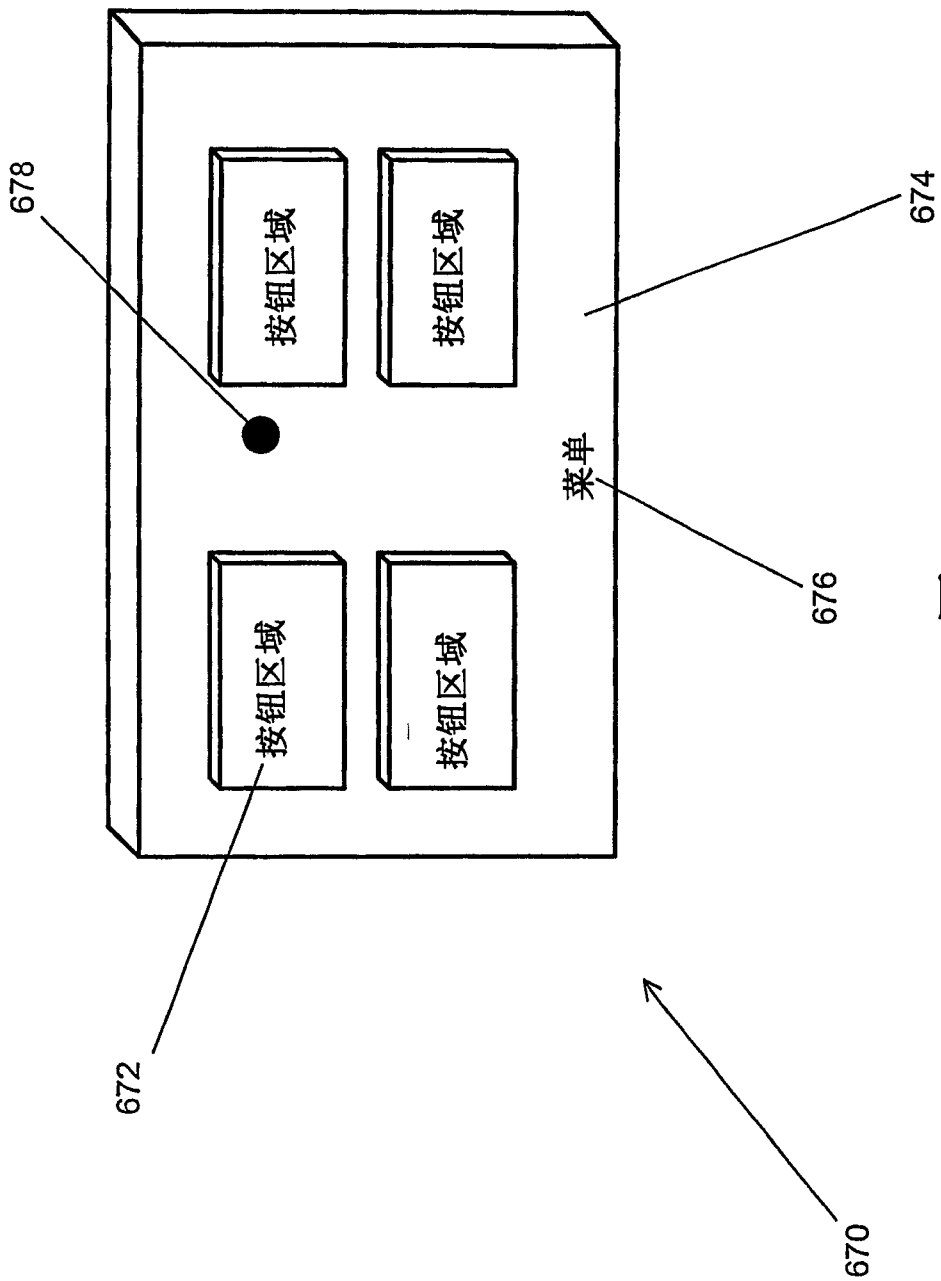


图4.2

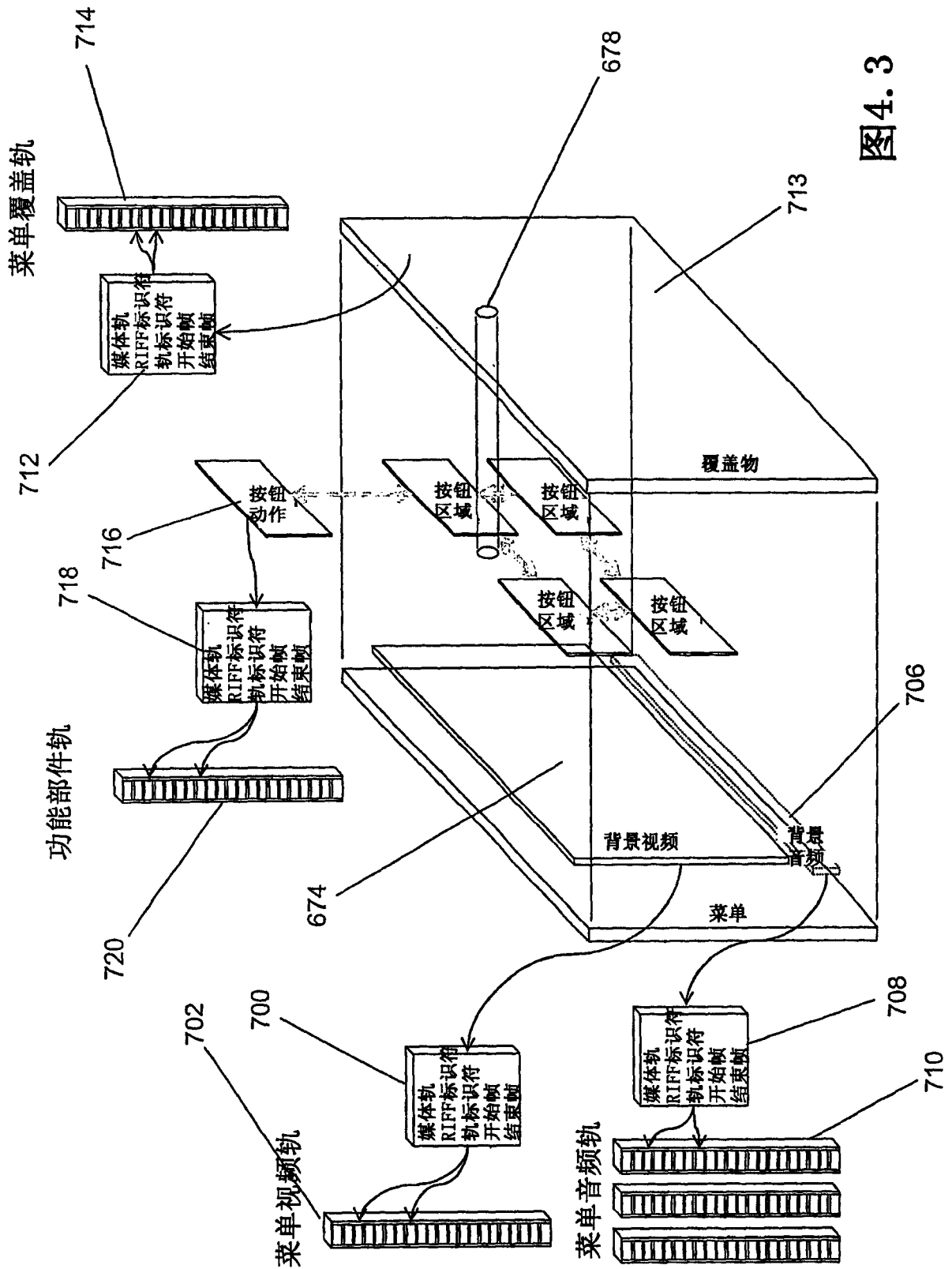


图4.3