THIS SHEET HAS BEEN REPLACED

DESCRIPTION

Apparatus and Method For Extracting Measures of a Bitstream's Processing Requirements For Decoding

Technical Field

This invention relates to appartus and method for extracting measures of a bitstream's processing requirements for decoding, and more particularly, to the field of digital audio and video decoders and the extraction of the processing requirements of a bitstream to be decoded by these decoders.

Background Art

Digital audio and video has become widespread in the field of consumer electronics, due in large part to the emergence of digital video standards such as MPEG1 (see "CD11172 - Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbps" by International Organisation for Standardisation, ISO MPEG Document, ISO-IEC/JTC1/SC2/WG11, 1994), MPEG2 (see "IS13818 - Generic coding of Moving Pictures and Associated Audio" by International Organisation for

This Sheet Has Been Replaced



THIS SHEET HAS BEEN REPLACED

Standardisation, ISO MPEG Document, ISO-IEC/JTC1/SC2/WG11, 1994), and the emerging standard MPEG4, which includes further added functionality and quality over MPEG1 and MPEG2. In MPEG1 and MPEG2 decoders, there are certain expectations regarding the decoder processing power which must be met in order to conform to the respective standard. For example, in the MPEG2 standard, Main Profile and Main Level (MP@ML) describes a set of bounds on the bitstream and associated video which all conformant MP@ML decoders must adhere to.

There are many situations where it is desirable to know the processing requirements of decoding a bitstream. For example, in a system which dynamically allocates resources for decoding, this information can be used during resource allocation to ensure the decoder reserves enough computational power and memory. In the design of integrated circuits for digital video decoders it has been common practice to determine the processing requirements for bitstreams based on the MP@ML guidelines. In other words, many integrated circuits were designed to allocate resources to handle the worst or near-worst case computational requirements for MPEG2 bitstreams. Another way to determine the processing requirements of a bitstream is to decode the bitstream and measure the

This Sheet Has Been Replaced



performance of the decoder. For example, one measurement of bitstream complexity could be the time taken to decode the bitstream. This was a very common method used by software based MPEG1 video decoders running on personal computers. If the decoder was too slow for synchronisation of audio and video output signals, the video decoder could skip some picture decoding. MPEG4 includes the concept of video objects, which are picture sequences which may be used to describe a separable object in a scene. If a scene contains many objects, it is desirable to know the objects processing requirements for decoding so that decoder resources can be efficiently allocated to objects. If there are multiple decoders in a system which use the same resources, it is desirable to know the decoder's processing requirements so that resources can be efficiently allocated to each decoder.

There is a growing abundance of digital audio, video and graphics information bitstreams available, and a growing number of processing requirements associated with each. It is desirable to know the processing requirements of a bitstream before it is decoded. This permits better control over the result of decoding. For example, if the decoder cannot meet the requirements dictated by the bitstream, it can have the option of not starting the decoding. For example, in a video decoder, if a picture

THIS SHEET HAS BEEN REPLACED

cannot be decoded in time to meet it's real time requirements, the picture decoding can be skipped. In an MPEG4 decoder, if an object cannot be decoded to meet the real time requirements, the object can be skipped. An object of this invention is to extract information from a bitstream which can be used to predict the processing requirements for decoding that bitstream.

In a system which dynamically allocates resources to a bitstream decoder, it is desirable to know if the resources are sufficient to permit real time decoding. In a video decoder, for example, it is desirable to know if resources are sufficient to complete the decoding of a picture so that it's display can occur as intended by the bitstream author. If the allocation is not appropriate, a resource reallocation can be performed. It is an object of this invention to permit the processing requirements of a bitstream to be measured in order for the efficient allocation of computational and memory resources to a decoder function, which can be used to help maintain decoder performance.

Disclosure Of Invention

For the purpose of solving the above described problems, the method of extracting measures of a bitstream's processing requirements for decoding described herein was invented. A parser parses a bitstream for

syntax elements. A processor processes said syntax elements and determines processing indicators. A buffer stores said indicators. Alternatively, a buffer stores said syntax elements and a processor processes said stored syntax elements and determines processing indicators. Said parser's position in the decoder may be before said decoder's input bitstream buffer. Said parser may be combined or be a subset of said decoder's bitstream syntax decoder which parses the bitstream for the purpose of decoding. Said parser and decoder's bitstream syntax decoder may be before or after said decoder's input bitstream buffer. Said processor processing may include analysing occurrences of said syntax elements to obtain some measures of processing requirements for decoding said bitstream. Said decoder may comprise of one or more video decoders, one or more audio decoders, one or more graphics decoders, or a combination of these decoders.

The parser extracts syntax elements useful for determining the processing requirements of decoding the bitstream. In some cases it may be possible to combine this function with the decoder's bitstream syntax decoder since it's function is also to extract syntax elements from the bitstream. If the parser and decoder's bitstream syntax decoder are located before the decoder's input bitstream buffer, the input bitstream buffer can store extracted syntax

THIS SHEET HAS BEEN REPLACED

elements for the decoding function to use later. The extracted syntax elements passed to the processor are processed in order to determine some processing indicators, which are measures of the processing requirements for decoding the bitstream. These extracted syntax elements may or may not be buffered before the processor operates on them. The processing indicators made by the processor may or may not be buffered. The buffer is used to permit the measures of processing requirements to be delayed until they are required.

Brief Description Of Drawings

Fig. 1 is a block diagram of a first mbodiment of the invention.

Fig. 2 is a block diagram of a second embodiment of the invention.

Fig. 3 is a block diagram of a third embodiment of the invention.

Fig. 4 is a block diagram of a fourth embodiment of the invention.

Best Mode for Carrying Out the Invention

A first embodiment of this invention is shown in Fig. 1. One or more bitstreams are input via the bitstream input 100, 101, 102. The decoder 110 decodes the bitstreams

This Sheet Has Been Replaced



THIS SHEET HAS BEEN REPLACED

and outputs decoded outputs 103. The parser 120 accepts the bitstreams 102 and outputs syntax elements 121. The processor 122 accepts the syntax elements 121 and outputs indicators 123. The buffer 124 accepts the indicators 123 and outputs measures of the bitstream's processing requirements 125.

The operation of the first embodiment in Fig. 1 is now described. The parser 120 extracts syntax elements relevant to determining the processing requirements of the bitstreams and passes them to the processor 122. The processor 122 determines from these syntax elements one or more indicators which can be used to measure the bitstreams processing requirements. These indicators are buffered by the buffer 124 until they are required.

Second embodiment of this invention is shown in Fig. 2. It has a similar structure to that of Fig. 1 except that the order of the processor 224 and buffer 222 is changed. The buffer 222 accepts the syntax elements 221 and outputs buffered syntax elements 223. The processor 224 accepts the buffered syntax elements 223 and outputs measures of the bitstream's processing requirements 225. In addition, more detail about the decoder is given. The bitstream input 201 is passed to the decoder's input buffer 211, which passes the buffered bitstream 212 to a decoder parser 213. The decoder parser passes decoder syntax elements 214 to

This Sheet Has Been Replaced



THIS SHEET HAS BEEN REPLACED

the simple decoder 215, which outputs decoded outputs 203.

The operation of the second embodiment in Fig. 2 is now described. The parser 220 extracts syntax elements relevant to determining the processing requirements of the bitstreams and passes them to the buffer 222 for storage. The processor 224 determines from these stored syntax elements one or more measures of the bitstreams processing requirements 225. The decoder input buffer 211 stores the input bitstream until the appropriate time for decoding. The decoder parser 213 performs some or all of the syntax decoding functions of the decoder 210. The simple decoder 215 performs the remaining functions of the decoder 210.

As described by Fig. 1 and 2, the location of the buffer can be before or after the processor. It is also possible to have buffers both before and after the processor.

The actual structure of the decoder 210 shown in Fig. 2 is not a limitation of this invention. This structure is used to illustrate the differences between the embodiment of Fig. 2 and the embodiment of Fig. 3.

In Fig. 3, showing the third embodiment, the parser 320 outputs decoder syntax elements 314 to a simple decoder 315, and outputs processing syntax elements 321 to the buffer 322. The buffer 322 accepts the processing syntax elements 321 and outputs buffered processing

This Sheet Has Been Replaced



syntax elements 323. The processor 324 accepts the buffered processing syntax elements 323 and outputs measures of the bitstream's processing requirements 325.

The operation of the third embodiment in Fig. 3 is now described. The parser 320 extracts processing syntax elements 321 relevant to determining the processing requirements of the bitstreams and passes them to the buffer 322 for storage, while also extracting decoder syntax elements 314 and passing them to the simple decoder 315. In this case a simple decoder may be simpler than the full decoder because some of it's syntax decoding functions are performed by the parser 320.

The effect of this embodiment is that parsing and decoder bitstream syntax decoding functions can be performed by the same block, thus reducing cost. Both functions require similar functions which can easily be shared, such as syntax understanding and syntax element extraction.

Fig. 4 shows a fourth embodiment in which, the parser 420 outputs decoder syntax elements 412 to a decoder input buffer 411, which outputs buffered decoder syntax elements 414 to the simple decoder 415. The parser also outputs processing syntax elements 421 to the buffer 422. The buffer 422 accepts the processing syntax elements 421 and outputs buffered processing syntax elements 423. The

processor 424 accepts the buffered processing syntax elements 423 and outputs measures of the bitstream's processing requirements 425.

The operation of the fourth embodiment in Fig. 4 is now described. The parser 420 extracts processing syntax elements 421 relevant to determining the processing requirements of the bitstreams and passes them to the buffer 422 for storage, while also extracting decoder syntax elements 414 and passing them to the decoder input buffer 411.

The effect of this embodiment is similar to that of the embodiment described by Fig. 3. Parsing and decoder bitstream syntax decoding functions can be performed by the same block, thus reducing cost. Both functions require similar functions which can easily be shared, such as syntax understanding and syntax element extraction. An additional effect is that the measures of processing requirements 425 may be determined much earlier because the parser 420 extracts the processing syntax elements before the decoder input buffer 411. This means that the dynamic allocation of system resources for decoding the bitstream can be done before the decoding.

In all of the embodiments described by Figs 1 through 4, the buffer 124, 222, 322, 422 is used to supply a delay determined by the embodiment's structure and the system's

requirements for the measures of bitstream's computational requirements 125, 225, 325, 425. For example, in Fig. 2 the decoder input buffer 211 may be the video buffer verifier input buffer of an MPEG2 video decoder. This buffer stores incoming bitstreams such that a picture can be instantaneously extracted and decoded, as described in the MPEG2 specification. This buffer can also be used to synchronize decoders by allowing decoding and corresponding outputs to be delayed. In the example of an MPEG2 video decoder, the complexity of the picture's coded bitstream can be determined before any of the picture is extracted from the decoder input buffer 211. Thus, the system resources required by the decoder 210 can be determined and allocated before the picture decoding begins. Since it may be convenient to assign resources at the start of decoding a new picture, it is convenient to have a picture's measures of bitstream's computational requirements 225 before the picture is decoded. Thus, the delay through the buffer 222 is similar to the delay of the decoder input buffer 211 for this example. This is not a limitation of the invention however, because other examples of this invention will require different buffer characteristics. For example, for a video decoder, the delay could vary from no delay to many pictures of delay. As an additional variation, some

information occurs in the bitstream with different frequency than others, thereby requiring different buffering.

This invention is not limited to any particular type of decoder. The decoder can be a video decoder, an audio decoder, a graphics decoder, or combinations and hybrids of these.

The parser may extract syntax elements from the bitstream which are used directly in the decoding process, or it may extract syntax elements that are placed in the bitstream for the purposes of indicating bitstream decoding computational requirements. This is not a limitation of the invention.

The processor processing is not limited by this invention. The extracted data can be simply counted, averaged, filtered, or any number of operations, including none, may be performed.

As another example embodiment, consider Fig. 2 again. In this example, the decoder 210 is an MPEG2 video decoder. The parser 220 extracts the following syntax elements 221 from the bitstream input 202:

(1) horizontal_size_value

(2) vertical_size_value

(3) frame_rate_code

(4) picture_coding_type (I, P, B,D)

These elements can be used to determine some

indication of the possible processing requirements of the bitstream. The picture size and rate can be determined from these elements.

These are not the only elements which can be used. The definition of the syntax elements used and how they are used is not a limitation of this invention. Other elements can be used which can further refine the estimation of the processing requirements for decoding the bitstream.

For example, the macroblock_type can be extracted and processed to determine the number of bi-directionally predicted macroblocks that are coded in a picture. This will help determine the computational requirements of decoding that picture. Bi-directionally predicted macroblocks require about twice the amount of frame memory accessing required by uni-directionally predicted macroblocks.

One effect of this invention is that information can be extracted from a bitstream which can be used to predict the processing requirements for decoding that bitstream. The parser extracts syntax elements relevant to determining the processing requirements. Parsing can be performed together with the decoder's existing syntax decoder or separately, and can occur before or after the decoder's input bitstream buffer. The processor processes these syntax elements resulting in measures which indicate the

processing requirements of decoding the bitstream. The buffer enables the measures to be available as and when needed. An effect of this invention is that the processing power requirements of a bitstream may be measured in order for the efficient allocation of processing resources to a decoder function, which can be used to help maintain decoder performance. Ultimately this results in better picture and audio quality.

This Sheet Has Been Replaced

THIS SHEET HAS BEEN REPLACED

CLAIMS

1. An apparatus for extracting measures of a bitstream's processing requirements for decoding, comprising:

parser means for parsing a bitstream for syntax elements;

processor means for processing said syntax elements and determining processing indicators;

buffer means for storing said indicators.

2. An apparatus for extracting measures of a bitstream's processing requirements for decoding comprising:

parser means for parsing a bitstream for syntax elements;

buffer means for storing said syntax elements;

processor means for processing said stored syntax elements and determining processing indicators.

3. An apparatus according to claim 1 or 2, wherein said parser comprises:

a bitstream syntax decoder which extracts syntax elements which are relevant for use by said processor from said bitstream before said bitstream is buffered in said decoder's input buffer.

* C L M S *

4. An apparatus according to claim 1 or 2, wherein said parser comprises:

said video decoder's bitstream syntax decoder which extracts syntax elements from said decoder's input buffer for bitstream decoding, and which also extracts syntax elements for use by said processor or storage by said buffer.

5. An apparatus according to claim 1 or 2, wherein said parser comprises:

said decoder's bitstream syntax decoder which extracts syntax elements from said bitstream for the purpose of bitstream decoding before buffering of extracted syntax elements in said decoder's input buffer, and which also extracts syntax elements for use by said processor or storage by said buffer.

6. An apparatus according to any one of claims 1 to 5, wherein said processing said syntax elements comprises:

means for analysing occurrences of said syntax elements to obtain some measures of processing requirements for decoding said bitstream.

7. An apparatus according to any one of claims 1 to 6, wherein said decoder comprises:

THIS SHEET HAS BEEN REPLACED

one or more video decoders, one or more audio decoders, one or more graphics decoders, or a combination of these decoders.

8. A method of extracting measures of a bitstream's processing requirements for decoding, comprising the steps of:

parsing a bitstream for syntax elements;

processing said syntax elements and determining processing indicators;

storing said indicators.

9. A method of extracting measures of a bitstream's processing requirements for decoding comprising:

parsing a bitstream for syntax elements;

storing said syntax elements;

processing said stored syntax elements and determining processing indicators.

10. A method according to claim 8 or 9, wherein said parsing comprises the steps of:

decoding said bitstream syntax for extracting syntax elements which are relevant for use by said processor from said bitstream before said bitstream is buffered in said decoder's input buffer.

THIS SHEET HAS BEEN REPLACED

11. A method according to claim 8 or 9, wherein said parsing comprises the steps of:

decoding said bitstream syntax for extracting syntax elements from said decoder's input buffer for bitstream decoding, and also for extracting syntax elements for use by said processor or storage by said buffer.

12. A method according to claim 1 or 2, wherein said parsing comprises the steps of:

decoding said bitstream syntax for extracting syntax elements from said bitstream for bitstream decoding before buffering of extracted syntax elements in said decoder's input buffer, and also for extracting syntax elements for use by said processor or storage by said buffer.

13. A method according to any one of claims 1 to 5, wherein said processing said syntax elements comprises the step of:

analysing occurrences of said syntax elements to obtain some measures of processing requirements for decoding said bitstream.

14. A method according to any one of claims 1 to 6, wherein said decoder comprises:

one or more video decoders, one or more audio decoders, one or more graphics decoders, or a combination of these decoders.

THIS SHEET HAS BEEN REPLACED