**EXHIBIT A-10**

**U.S. Patent No. 5,650,783 ("Murashita")**

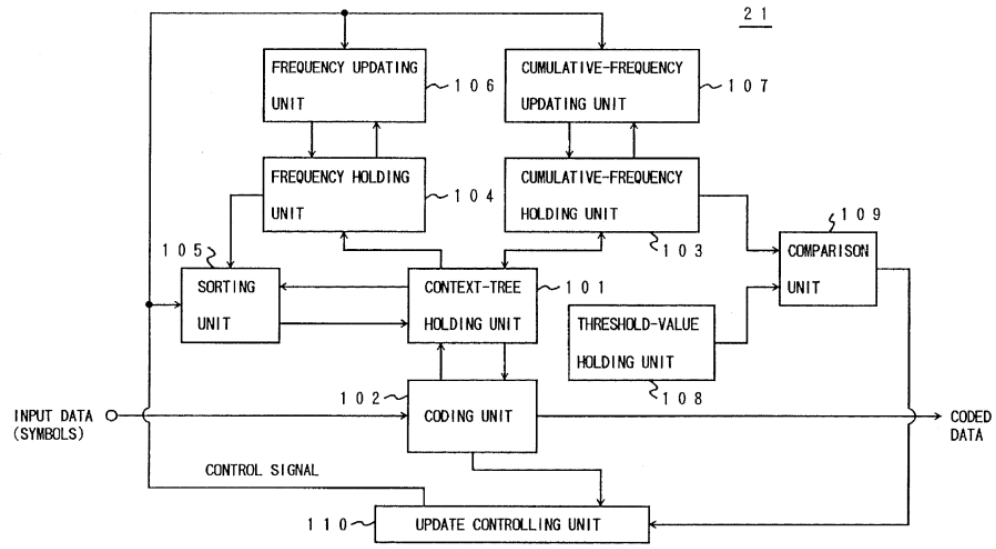
**ANTICIPATES / RENDERS OBVIOUS U.S. PATENT NO. 9,036,701**

As described in the following claim chart, Asserted Claims 1, 3–4, 6–7, 8, 10–11, 13–14, 15, 17–18, and 20 of U.S. Patent No. 9,036,701 (the "'701 patent") are invalid in view of US Patent No. 5,650,783 ("Murashita"), entitled Data coding/decoding device and method, issued on July 22, 1997. Murashita was filed on September 8, 1995, and is prior art under at least 35 U.S.C. §§ 102 (a), (b), and (e).

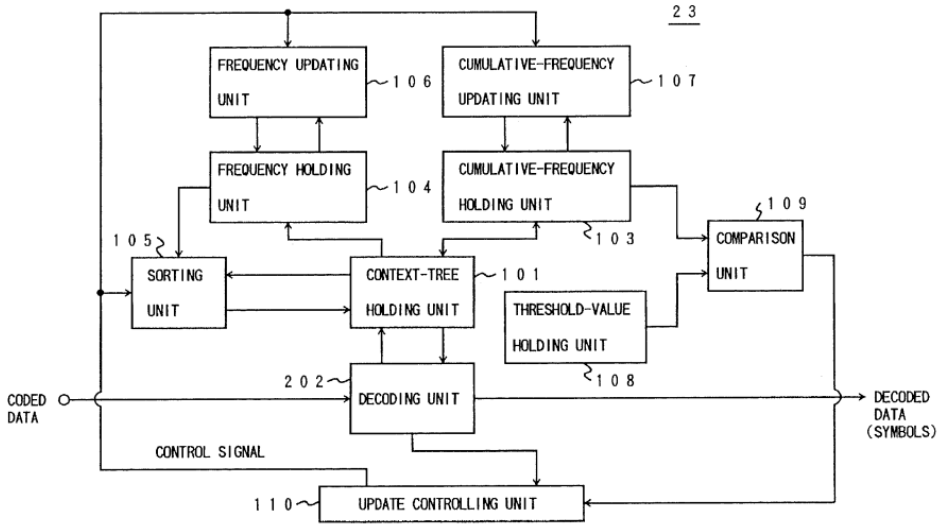
To the extent Complainants allege that Murashita alone does not disclose any particular limitation of the Asserted Claims of the '701 patent, either expressly or inherently, it would have been obvious to a person of ordinary skill in the art as of the priority date of '701 patent to modify Murashita and/or to combine the teachings of Murashita with: (1) the knowledge of one of ordinary skill in the art to show all the limitations of the claim; (2) the teachings of the prior art references set forth in any of the references identified in Exhibit A-11.

To the extent that these Invalidity Contentions rely on or otherwise embody particular constructions of terms or phrases in the Asserted Claims, Respondents are not proposing any such constructions as proper constructions of those terms or phrases. Various positions put forth in this document are predicated on Complainants' incorrect and overly broad interpretation of its claims. Those positions are not intended to and do not necessarily reflect Respondents' interpretation of the true and proper scope of Complainants' claims, and Respondents reserve the right to adopt claim construction positions that differ from or even conflict with various positions put forth in this document.

| **ASSERTED CLAIMS** | **PRIOR ART REFERENCE:**<br>**Murashita** |
|---|---|
| 1. [Pre] A method comprising: | Murashita teaches and/or discloses a method comprising the following limitation:<br><br>*See, e.g.*, Murashita at Title: "Data coding/decoding device and method."<br><br>Murashita at Fig. 9 and associated description. |




Murashita at Fig. 11 and associated description.

| | |
|---|---|
| |  <br>*See also* Murashita at Figs. 13, 15 and associated descriptions. |
| [a] determining a frequency of occurrence threshold based on an expected frequency of occurrence of syntax elements in a bit stream; | Murashita teaches and/or discloses determining a frequency of occurrence threshold based on an expected frequency of occurrence of syntax elements in a bit stream:<br><br>*See, e.g.*, Murashita at 6:40-47.<br><br>> The controlling unit 13 controls the context holding/updating unit 12 to update the context data. In doing so, the controlling unit 13 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 13 controls the context holding/updating unit 12 to update the context data for the given context.<br><br>Murashita at 6:57-64. |

| | |
|---|---|
| | The controlling unit 16 controls the context holding/updating unit 15 to update the context data. In doing so, the controlling unit 16 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 16 controls the context holding/updating unit 15 to update the context data for the given context.<br><br>Murashita at 7:40-57.<br><br>The threshold-value holding unit 108 stores a threshold value which is used in making a decision to stop the updating of the frequency, the cumulative frequency, and the context tree. The comparison unit 109 compares the threshold value stored in the threshold-value holding unit 108 with the cumulative frequency stored in the cumulative-frequency holding unit 103. Then, the comparison unit 109 outputs a comparison-output signal, which is a low level signal when the cumulative frequency is less than the threshold value, and which is a high level when the cumulative frequency is not less than the threshold value. The update controlling unit 110 controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 when a new symbol is supplied to the coding unit 102. In doing so, the update controlling unit 110 uses the comparison-output signal supplied thereto for determining whether to initiate the updating processes of the frequency, the cumulative frequency, and the context tree.<br><br>Murashita at 12:13-41.<br><br>The description of the embodiment in the above has been given with regard to a case in which only one threshold value is used, i.e., a threshold value is the same for any context. However, a different threshold value may be provided for each context. This is useful because the number of the symbols registered under a given context is dependent on the given context. In English text, for example, a large number of symbols are registered under such contexts as " " (space), ", " (comma), ". " (period), etc. By providing different threshold values according to the frequency of occurrence, degradation of the compression rate can be minimized. |

| | |
|---|---|
| | Since the use of a different threshold value for each context leads to a complexity of the coding process, a different threshold value can be provided for each order of contexts. Despite the dependency on the contexts, a smaller number of symbols are registered under a context of a higher order. This means that a longer processing time is required for the updating process of the smaller order. Thus, preparing a different threshold value for each order can bring about an efficient processing. Especially, under contexts of the zero-th order, all the symbols are registered ranging from 0x00 to 0xFF plus a special code EOF (end of file). The processing time of the zero-th order is far larger than those of other orders. Thus, the control of the updating process according to the present invention can be very effective even if it is applied only to the contexts of the zero-th order. In this case, the control of the entire process is simple, and an increase in circuit complexity is minimized. |
| [b] categorizing a plurality of syntax elements of video content into first and second categories based on the frequency of occurrence threshold, wherein syntax elements which occur greater than the frequency of occurrence threshold are categorized into the second category and syntax elements which occur less than the frequency of occurrence are categorized into the first category;[1] | Murashita teaches and/or discloses categorizing a plurality of syntax elements of video content into first and second categories based on the frequency of occurrence threshold, wherein syntax elements which occur greater than the frequency of occurrence threshold are categorized into the second category and syntax elements which occur less than the frequency of occurrence are categorized into the first category:<br><br>*See, e.g.*, Murashita at 6:40-47.<br><br>The controlling unit 13 controls the context holding/updating unit 12 to update the context data. In doing so, the controlling unit 13 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 13 controls the context holding/updating unit 12 to update the context data for the given context.<br><br>Murashita at 6:57-64. |

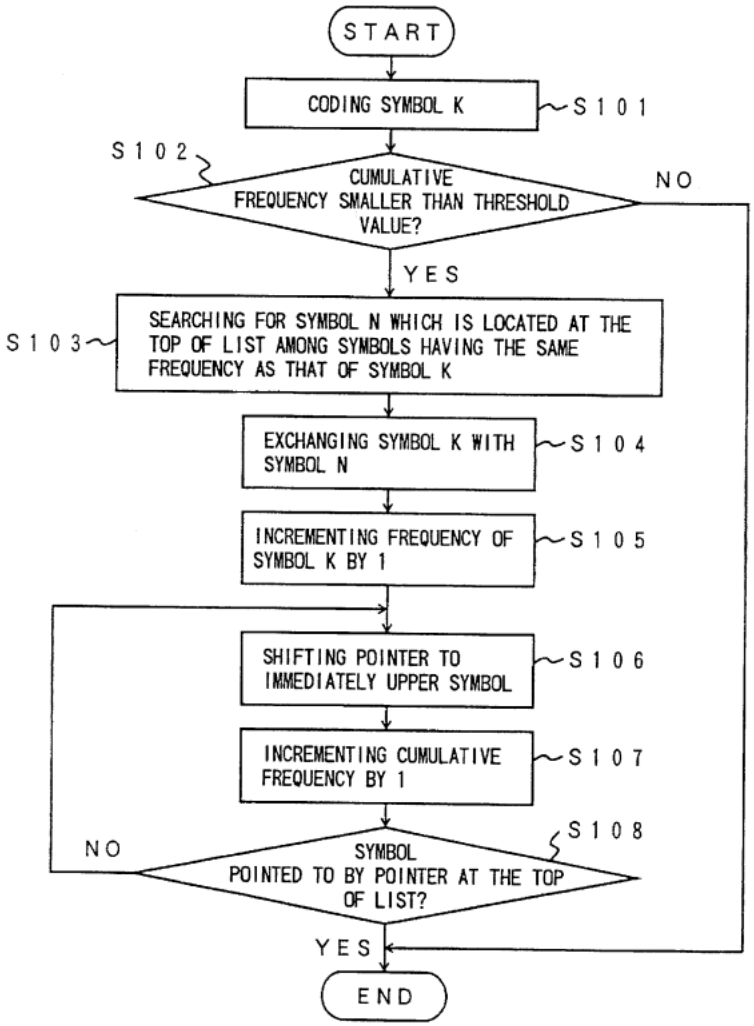
[1] For the convenience and clarity, the corrections from the certificate of correction have been incorporated into this limitation and the corresponding limitations of claims 8 and 15, below.

| | |
|---|---|
| | The controlling unit 16 controls the context holding/updating unit 15 to update the context data. In doing so, the controlling unit 16 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 16 controls the context holding/updating unit 15 to update the context data for the given context. |
| [c] entropy coding symbols that correspond to the first category of syntax elements and that have been subjected to a context update; and | Murashita teaches and/or discloses entropy coding symbols that correspond to the first category of syntax elements and that have been subjected to a context update:<br><br>*See, e.g.*, Murashita at Abstract.<br><br>A device for coding a series of symbols based on a multi-level arithmetic coding scheme includes a coding unit for coding the symbols based on frequency of occurrence of the symbols and orders of the symbols according to the frequency of occurrence, a context holding unit for storing the frequency of occurrence and the orders, a context updating unit for updating the frequency of occurrence and the orders upon receiving a control signal, and a controlling unit for sending the control signal when one of the symbols is coded and a cumulative frequency is less than a predetermined value, wherein the cumulative frequency is a sum of the frequency of occurrence for all of the symbols.<br><br>Murashita at 1:34-48.<br><br>Methods of compressing text data, files, etc. include a dictionary-type coding method which utilizes analogies in data series, and a statistical-type coding method which utilizes occurrence frequency of data-series. A typical method for the statistical-type coding is an arithmetic coding method. This method is regarded as being capable of coding data with the maximum efficiency when occurrence frequency for data from an information source can be known. This method does not encode each character one by one as does the Huffman coding method, but treats a character string in its entirety to |

| | |
|---|---|
| | boost an efficiency in data compression. The term "arithmetic coding" comes from the fact that coded words are derived through calculation and represented by binary numbers including numbers below a decimal point.<br><br>Murashita at 1:63-2:4.<br><br>Methods of compressing text data, files, etc. include a dictionary-type coding method An example of the multi-level arithmetic coding scheme is shown in FIGS. 1A and 1B. As shown in FIG. 1B, when events of characters (hereinafter called symbols) are to be encoded, a range P ($0 \leqq P < 1$) (hereinafter referred to as [0,1)) is divided by the number of the symbols. Here, a length of a divided range, which corresponds to one of the symbols, is in proportion to frequency of occurrence of the symbol. The frequency of occurrence is shown in FIG. 1A for symbols "a", "b", "c", "d", "e".<br><br>Murashita at 6:40-47.<br><br>The controlling unit 13 controls the context holding/updating unit 12 to update the context data. In doing so, the controlling unit 13 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 13 controls the context holding/updating unit 12 to update the context data for the given context.<br><br>Murashita at 6:57-64.<br><br>The controlling unit 16 controls the context holding/updating unit 15 to update the context data. In doing so, the controlling unit 16 compares a predetermined threshold value with a cumulative frequency of occurrence for all the symbols registered under a given context. Then, only when the cumulative frequency is smaller than the threshold value, the controlling unit 16 controls the context holding/updating unit 15 to update the context data for the given context.<br><br>Murashita at 7:40-57. |

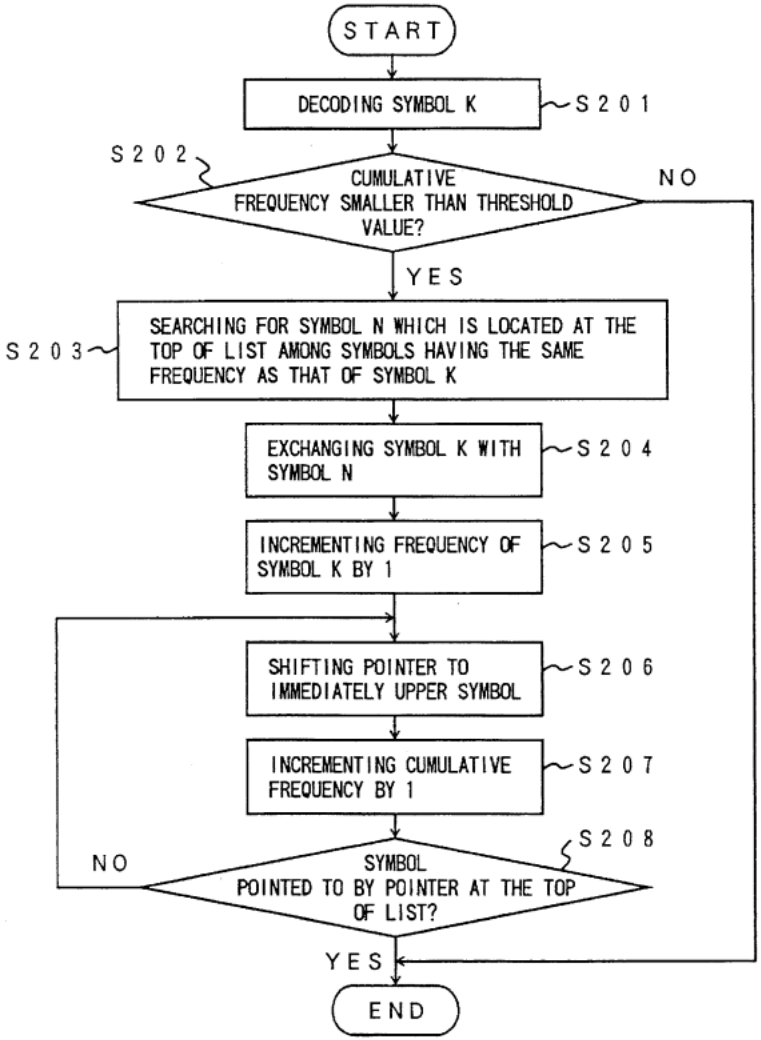
| | |
|---|---|
| | The threshold-value holding unit 108 stores a threshold value which is used in making a decision to stop the updating of the frequency, the cumulative frequency, and the context tree. The comparison unit 109 compares the threshold value stored in the threshold-value holding unit 108 with the cumulative frequency stored in the cumulative-frequency holding unit 103. Then, the comparison unit 109 outputs a comparison-output signal, which is a low level signal when the cumulative frequency is less than the threshold value, and which is a high level when the cumulative frequency is not less than the threshold value. The update controlling unit 110 controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 when a new symbol is supplied to the coding unit 102. In doing so, the update controlling unit 110 uses the comparison-output signal supplied thereto for determining whether to initiate the updating processes of the frequency, the cumulative frequency, and the context tree.<br><br>Murashita at 8:11-19.<br><br>The comparison unit 109 compares the threshold value stored in the threshold-value holding unit 108 with the cumulative frequency regarding all the symbols listed under the pertinent context. The comparison unit 109 supplies the comparison-output signal to the update controlling unit 110. When the cumulative frequency is less than the threshold value, the update controlling unit 110 controls the frequency updating unit 106, the cumulative-frequency updating unit 107, and the sorting unit 105 to carry out a respective updating process.<br><br>Murashita at 8:35-48.<br><br>At a step S101, a symbol K is coded by the coding unit 102. At a step S102, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S103. If it is not, the procedure ends.<br>Here, a result of the check is indicated by the level of the comparison-output signal generated by the comparison unit 109. A low level signal indicates that the cumulative |

| | |
|---|---|
| | frequency is less than the threshold value. A high level signal indicates the contrary. Upon receiving the low level comparison-output signal, the update controlling unit 110 controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 to conduct the updating process.<br><br>Murashita at Fig. 10. |



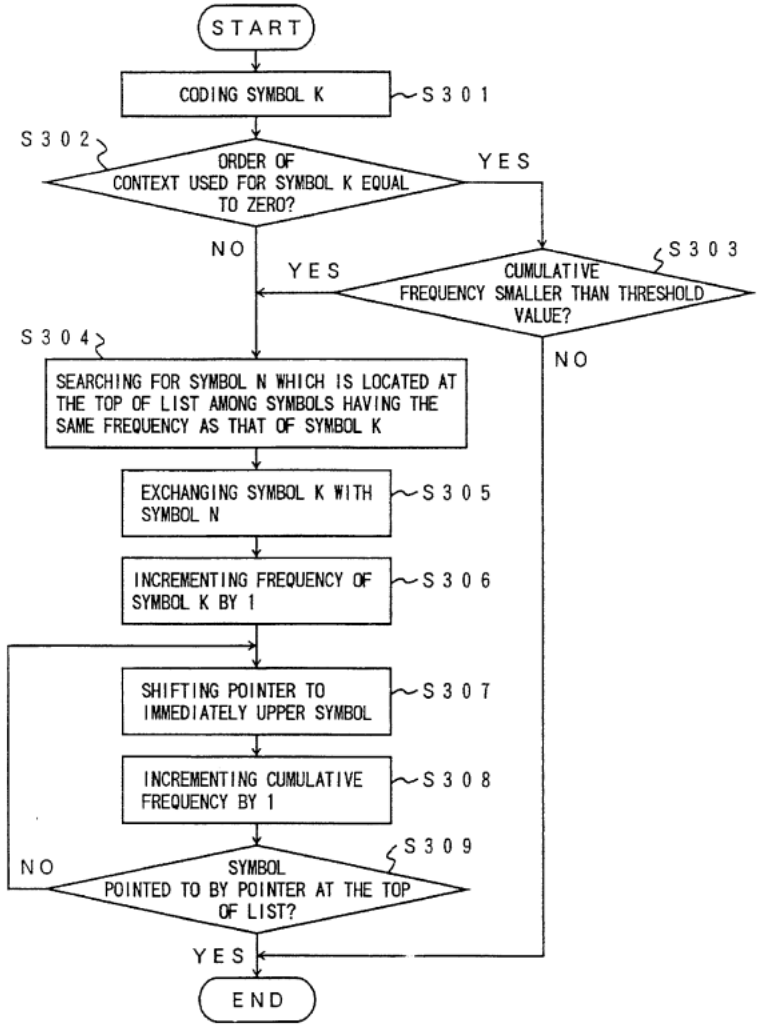

Murashita at 9:35-48.

| | |
|---|---|
| | At a step S201, a symbol K is decoded by the decoding unit 202. At a step S202, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S203. If it is not, the procedure ends.<br>Here, a result of the check is indicated by the level of the comparison-output signal generated by the comparison unit 109. A low level signal indicates that the cumulative frequency is less than the threshold value. A high level signal indicates the contrary. Upon receiving the low-level comparison-output signal, the update controlling unit 110 controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 to conduct the updating process.<br><br>Murashita at Fig. 12. |



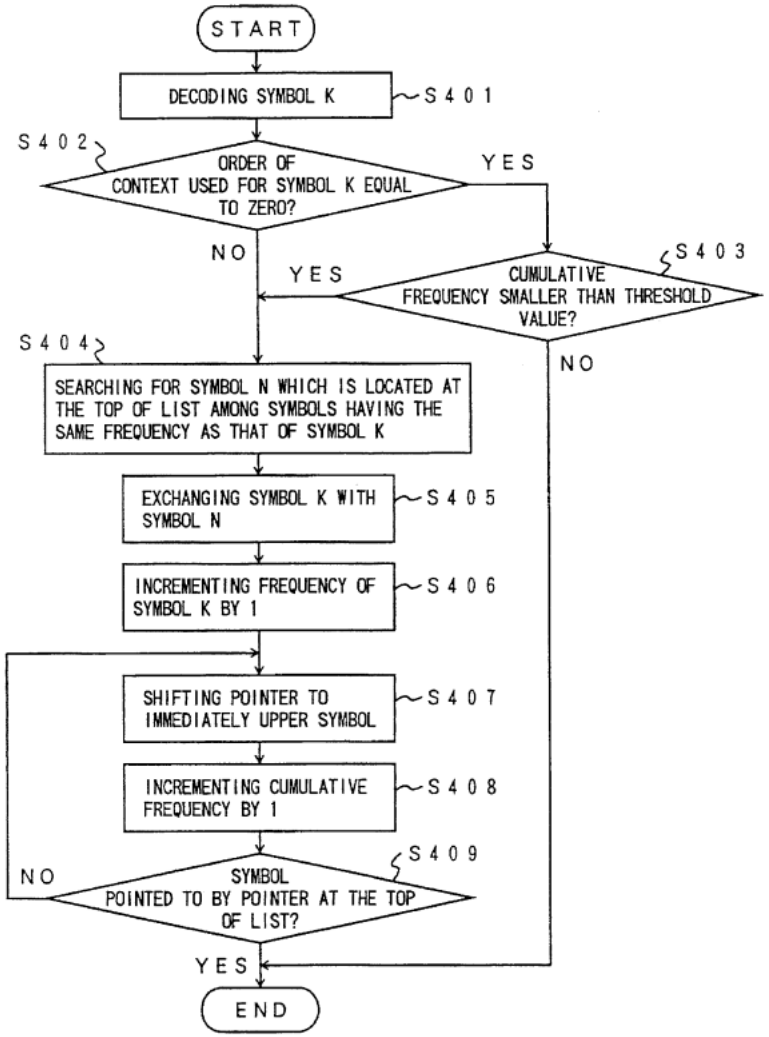

Murashita at 10:32-45.

| | |
|---|---|
| | At a step S303, a check is made by the comparison unit 109 whether the cumulative frequency is smaller than the threshold value. If it is, the procedure proceeds to a step S304. If it is not, the procedure ends.<br>Here, a result of the check at the step S303 is indicated by the level of the comparison-output signal generated by the comparison unit 109. A low level signal indicates that the cumulative frequency is less than the threshold value. A high level signal indicates the contrary. Upon receiving the low level comparison-output signal, the update controlling unit 110A controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 to conduct the updating process.<br><br>Murashita at Fig. 14. |



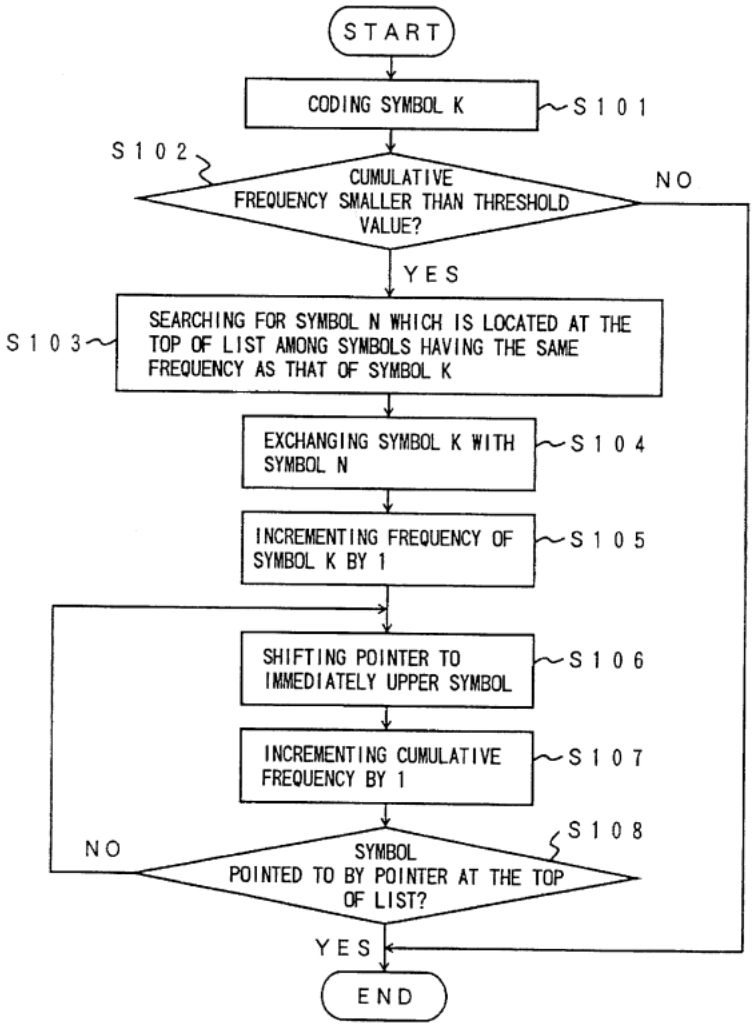

Murashita at 11:36-48.

| | |
|---|---|
| | At a step S403, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S404. If it is not, the procedure ends.<br>Here, a result of the check at the step S403 is indicated by the level of the comparison-output signal generated by the comparison unit 109. A low level signal indicates that the cumulative frequency is less than the threshold value. A high level signal indicates the contrary. Upon receiving the low level comparison-output signal, the update controlling unit 110A controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 to conduct the updating process.<br><br>Murashita at Fig. 16. |

| | |
|---|---|
| |  |
| [d] entropy coding symbols that correspond to the second category | Murashita teaches and/or discloses entropy coding symbols that correspond to the second category of syntax elements and that have bypassed context updating. |

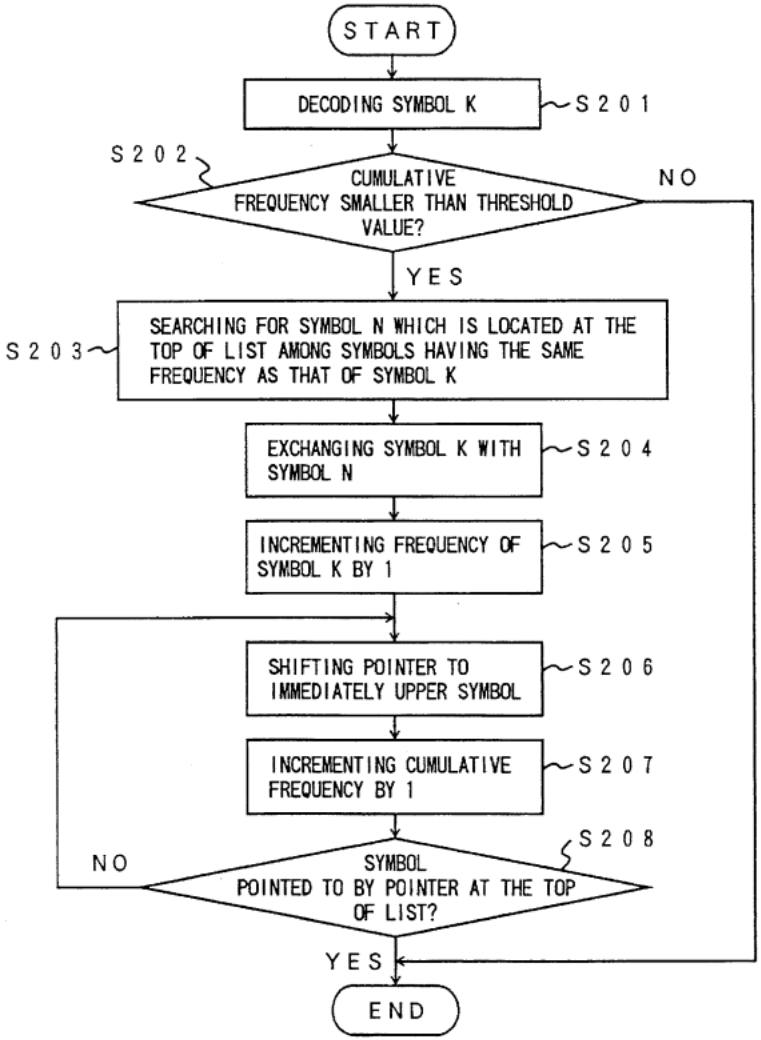
| | |
|---|---|
| of syntax elements and that have bypassed context updating. | *See, e.g.*, Murashita at 7:40-57.<br><br>The threshold-value holding unit 108 stores a threshold value which is used in making a decision to stop the updating of the frequency, the cumulative frequency, and the context tree. The comparison unit 109 compares the threshold value stored in the threshold-value holding unit 108 with the cumulative frequency stored in the cumulative-frequency holding unit 103. Then, the comparison unit 109 outputs a comparison-output signal, which is a low level signal when the cumulative frequency is less than the threshold value, and which is a high level when the cumulative frequency is not less than the threshold value. The update controlling unit 110 controls the sorting unit 105, the frequency updating unit 106, and the cumulative-frequency updating unit 107 when a new symbol is supplied to the coding unit 102. In doing so, the update controlling unit 110 uses the comparison-output signal supplied thereto for determining whether to initiate the updating processes of the frequency, the cumulative frequency, and the context tree.<br><br>Murashita at 8:21-32.<br><br>The sorting unit 105 searches for a symbol which is located at the top of the list among symbols having the same frequency of occurrence as that of the coded symbol (before an increment). Then, the sorting unit 105 exchanges the searched symbol with the coded symbol. The frequency updating unit 106 increments the frequency of occurrence for the coded symbol by 1. The cumulative-frequency updating unit 107 increments the cumulative frequency of all symbols listed above the coded symbol by 1. When the comparison-output signal indicates that the cumulative frequency is not less than the threshold value, the updating process described above will not take place.<br><br>Murashita at 8:35-39.<br><br>At a step S101, a symbol K is coded by the coding unit 102. At a step S102, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S103. If it is not, the procedure ends. |

| | |
|---|---|
| | Murashita at 9:1-6.<br><br>As mentioned, if it turns out at the step S102 that the cumulative frequency is not less than the threshold value, the procedure ends at that point. This is because the update controlling unit 110 receives the high level comparison-output signal, and decides that the updating process is no longer necessary.<br><br>Murashita at Fig. 10. |



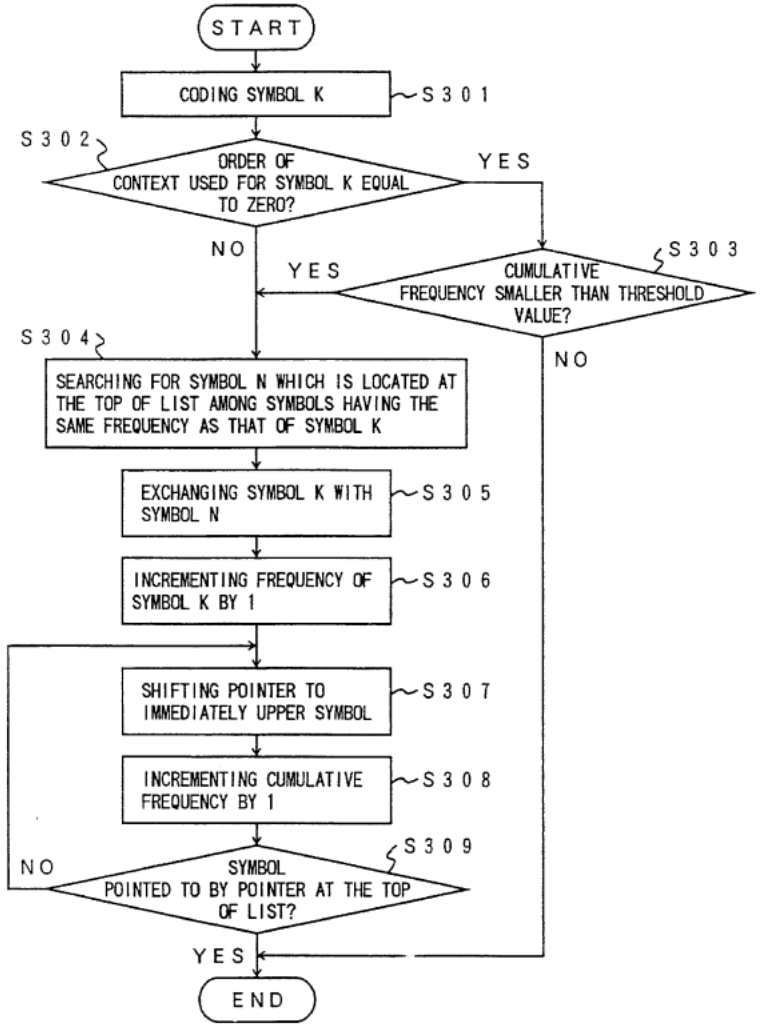

Murashita at 9:35-39.

| | |
|---|---|
| | At a step S201, a symbol K is decoded by the decoding unit 202. At a step S202, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S203. If it is not, the procedure ends.<br><br>Murashita at 10:1-6.<br><br>As mentioned, if it turns out at the step S202 that the cumulative frequency is not less than the threshold value, the procedure ends at that point. This is because the update controlling unit 110 receives the high level comparison-output signal, and decides that the updating process is no longer necessary.<br><br>Murashita at Fig. 12. |



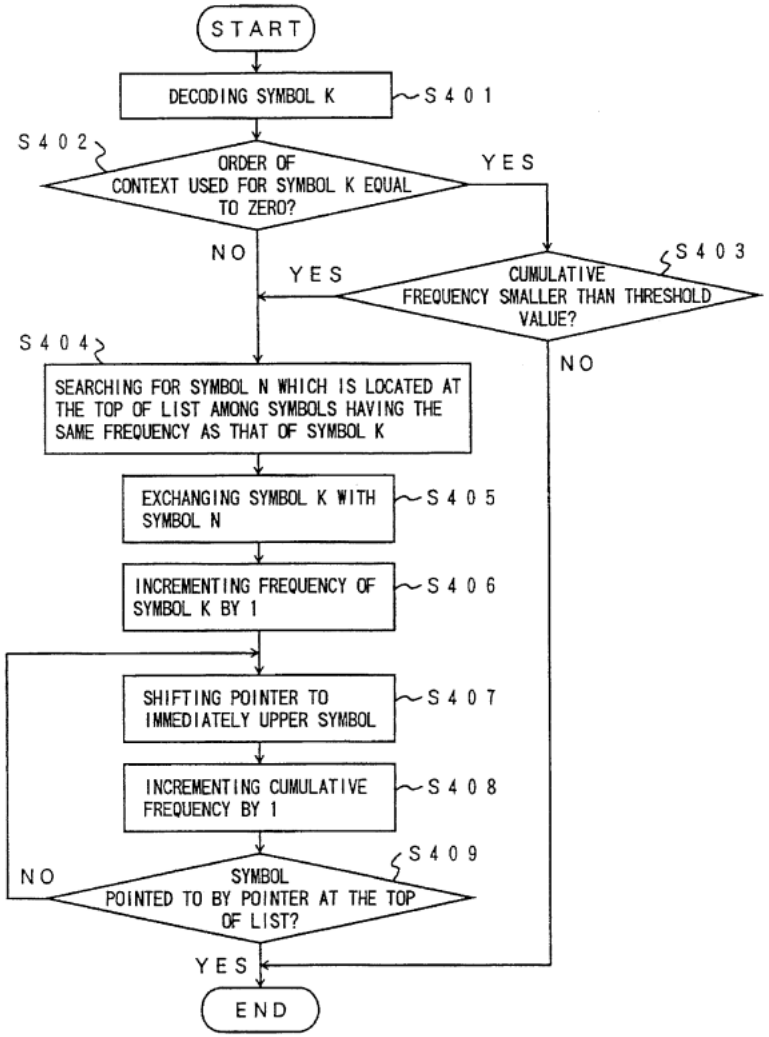

Murashita at 10:32-36.

| | |
|---|---|
| | At a step S303, a check is made by the comparison unit 109 whether the cumulative frequency is smaller than the threshold value. If it is, the procedure proceeds to a step S304. If it is not, the procedure ends.<br><br>Murashita at 10:65-11:3.<br><br>As mentioned, if it turns out at the step S303 that the cumulative frequency is not less than the threshold value, the procedure ends at that point. This is because the update controlling unit 110A receives the high level comparison-output signal, and decides that the updating process is no longer necessary.<br><br>Murashita at Fig. 14. |




Murashita at 11:36-39.

| | |
|---|---|
| | At a step S403, a check is made by the comparison unit 109 whether the cumulative frequency is less than the threshold value. If it is, the procedure proceeds to a step S404. If it is not, the procedure ends.<br><br>Murashita at 12:1-6.<br><br>As mentioned, if it turns out at the step S403 that the cumulative frequency is not less than the threshold value, the procedure ends at that point. This is because the update controlling unit 110A receives the high level comparison-output signal, and decides that the updating process is no longer necessary.<br><br>Murashita at Fig. 16. |



| | |
|---|---|
| 3. A method according to claim 1 further comprising performing binarization of the symbols that correspond to the second category of syntax elements prior to bypassing the context updating. | Murashita teaches and/or discloses a method according to claim 1 further comprising performing binarization of the symbols that correspond to the second category of syntax elements prior to bypassing the context updating.<br><br>*See, e.g.,* Murashita at 2:21-24.<br><br>Compressed codes for the character string "abe" are binary codes which represent an arbitrary point within the finally selected range. Through the same procedure, compressed codes for any character string can be obtained. |
| 4. A method according to claim 3 wherein performing binarization comprises performing binarization so as to target a predetermined probability. | Murashita teaches and/or discloses a method according to claim 3 wherein performing binarization comprises performing binarization so as to target a predetermined probability.<br><br>*See, e.g.,* Murashita at 2:21-24.<br><br>Compressed codes for the character string "abe" are binary codes which represent an arbitrary point within the finally selected range. Through the same procedure, compressed codes for any character string can be obtained. |
| 6. A method according to claim 1 wherein categorizing the plurality of syntax elements comprises categorizing the plurality of syntax elements based on a relationship of the frequency of occurrence of the syntax elements to a predefined threshold. | Murashita teaches and/or discloses a method according to claim 1 wherein categorizing the plurality of syntax elements comprises categorizing the plurality of syntax elements based on a relationship of the frequency of occurrence of the syntax elements to a predefined threshold.<br><br>*See* limitation 1[b] above, which is incorporated herein by reference |

| | |
|---|---|
| 7. A method according to claim 1 further comprising performing binarization of the symbols that correspond to the first category of syntax elements prior to entropy coding the symbols that correspond to the first category of syntax elements. | Murashita teaches and/or discloses A method according to claim 1 further comprising performing binarization of the symbols that correspond to the first category of syntax elements prior to entropy coding the symbols that correspond to the first category of syntax elements.<br><br>*See, e.g.*, Murashita at 2:21-24.<br><br>Compressed codes for the character string "abe" are binary codes which represent an arbitrary point within the finally selected range. Through the same procedure, compressed codes for any character string can be obtained. |
| 8. [pre] An apparatus comprising at least one processor and at least one memory including computer program code, the at least one memory and the computer program code configured to, with the processor, cause the apparatus to at least: | *See* claim 1 [pre] above, which is incorporated herein by reference.<br><br>*See also, e.g.*, Murashita at 1:8-13.<br><br>The present invention generally relates to data coding methods and devices, and particularly relates to a method and a device which encodes input data according to a variable-length coding scheme based on frequency of data-series occurrence in past data, and decodes coded data based on a history of past decoded data.<br><br>Murashita at 5:11-13.<br><br>Accordingly, there is a need in the field of coding/decoding schemes for a coding/decoding method and a device which have a faster processing speed. |
| [a] determine a frequency of occurrence threshold based on an expected frequency of occurrence of syntax elements in a bit stream; | *See* limitation 1[a] above, which is incorporated herein by reference. |

| | |
|---|---|
| [b] categorize a plurality of syntax elements of video content into first and second categories based on the frequency of occurrence threshold, wherein syntax elements which occur greater than the frequency of occurrence threshold are categorized into the second category and syntax elements which occur less than the frequency of occurrence are categorized into the first category; | *See* limitation 1[b] above, which is incorporated herein by reference. |
| [c] entropy code symbols that correspond to the first category of syntax elements and that have been subjected to a context update; and | *See* limitation 1[c] above, which is incorporated herein by reference. |
| [d] entropy code symbols that correspond to the second category of syntax elements and that have bypassed context updating. | *See* limitation 1[d] above, which is incorporated herein by reference. |
| 10. An apparatus according to claim 8 wherein the at least one memory and the computer program code are further configured to, with the processor, cause the apparatus to perform binarization of the symbols that correspond to the second category of syntax elements prior to bypassing the context updating. | *See* claim 3 above, which is incorporated herein by reference. |

| | |
|---|---|
| 11. An apparatus according to claim 10 wherein the at least one memory and the computer program code are configured to, with the processor, cause the apparatus to perform binarization by performing binarization so as to target a predetermined probability. | *See* claim 4 above, which is incorporated herein by reference. |
| 13. An apparatus according to claim 8 wherein the at least one memory and the computer program code are configured to, with the processor, cause the apparatus to categorize the plurality of syntax elements by categorizing the plurality of syntax elements based on a relationship of the frequency of occurrence of the syntax elements to a predefined threshold. | *See* claim 6 above, which is incorporated herein by reference. |
| 14. An apparatus according to claim 8 wherein the at least one memory and the computer program code are further configured to, with the processor, cause the apparatus to perform binarization of the symbols that correspond to the first category of syntax elements prior to entropy coding the symbols that correspond to the first category of syntax elements. | *See* claim 7 above, which is incorporated herein by reference. |

| | |
|---|---|
| 15. [pre] A computer program product comprising at least one non-transitory computer-readable storage medium having computer-executable program code instructions stored therein, the computer-executable program code instructions comprising program code instructions to: | *See* claim 1 [pre] above, which is incorporated herein by reference.<br><br>*See also, e.g.*, Murashita at 12:60-13:8.<br><br>1. A device for coding a series of symbols based on a multi-level arithmetic coding scheme, said device comprising:<br>coding means for coding said symbols based on frequency of occurrence of said symbols and orders of said symbols according to said frequency of occurrence;<br>context holding means for storing said frequency of occurrence and said orders;<br>context updating means for updating said frequency of occurrence and said orders upon receiving a control signal; and<br>controlling means for sending said control signal when one of said symbols is coded and a cumulative frequency is less than a predetermined value, wherein said cumulative frequency is a sum of said frequency of occurrence for all of said symbols.<br><br>Murashita at 13:32-49.<br><br>6. A device for decoding coded data to obtain a series of symbols based on a multi-level arithmetic decoding scheme, said device comprising:<br>decoding means for decoding said coded data to obtain said symbols based on frequency of occurrence of said symbols and orders of said symbols according to said frequency of occurrence;<br>context holding means for storing said frequency of occurrence and said orders;<br>context updating means for updating said frequency of occurrence and said orders upon receiving a control signal; and<br>controlling means for sending said control signal when one of said symbols is obtained and a cumulative frequency is less than a predetermined value, wherein said cumulative frequency is a sum of said frequency of occurrence for all of said symbols. |

| | |
|---|---|
| [a] determine a frequency of occurrence threshold based on an expected frequency of occurrence of syntax elements in a bit stream; | *See* limitation 1[a] above, which is incorporated herein by reference. |
| [b] categorize a plurality of syntax elements of video content into first and second categories based on the frequency of occurrence threshold, wherein syntax elements which occur greater than the frequency of occurrence threshold are categorized into the second category and syntax elements which occur less than the frequency of occurrence are categorized into the first category; | *See* limitation 1[b] above, which is incorporated herein by reference. |
| [c] entropy code symbols that correspond to the first category of syntax elements and that have been subjected to a context update; and | *See* limitation 1[c] above, which is incorporated herein by reference. |
| [d] entropy code symbols that correspond to the second category of syntax elements and that have bypassed context updating. | *See* limitation 1[d] above, which is incorporated herein by reference. |
| 17. A computer program product according to claim 15 wherein the computer-executable program code instructions further comprise program code instructions to perform binarization of the symbols that correspond to the second category of syntax elements prior to bypassing the context updating. | *See* claim 3 above, which is incorporated herein by reference. |

| | |
|---|---|
| 18. A computer program product according to claim 17 wherein the comprise program code instructions to perform binarization comprise program code instructions to perform binarization so as to target a predetermined probability. | *See* claim 4 above, which is incorporated herein by reference. |
| 20. A computer program product according to claim 15 comprise the program code instructions to categorize the plurality of syntax elements comprise program code instructions to categorize the plurality of syntax elements based on a relationship of the frequency of occurrence of the syntax elements to a predefined threshold. | *See* claim 6 above, which is incorporated herein by reference. |