# EXHIBIT E-6

# KARCZEWICZ I

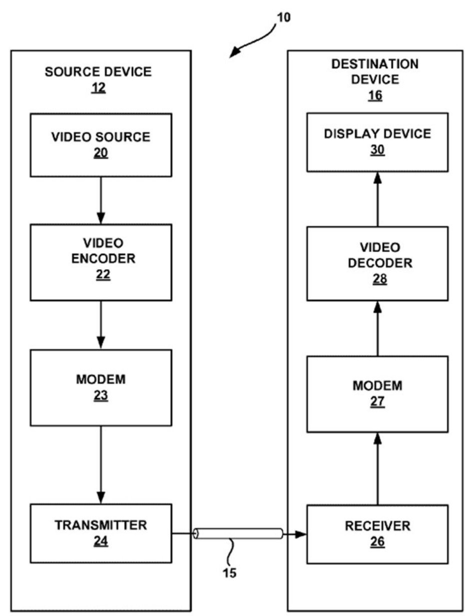
# ANTICIPATES / RENDERS OBVIOUS U.S. PATENT NO. 11,805,267

As described in the following claim chart, claims 1-36 of U.S. Patent No. 11,805,267 (the "'267 patent") are invalid in view of US Patent Application Publication 2011/0007799 ("Karczewicz I") (RESPNOKIA_00024739), entitled "Non-Zero Rounding and Prediction Mode Selection Techniques in Video Encoding," published on January 13, 2011. Karczewicz I was filed July 9, 2009, and is prior art under at least 35 U.S.C. § 102 (e).

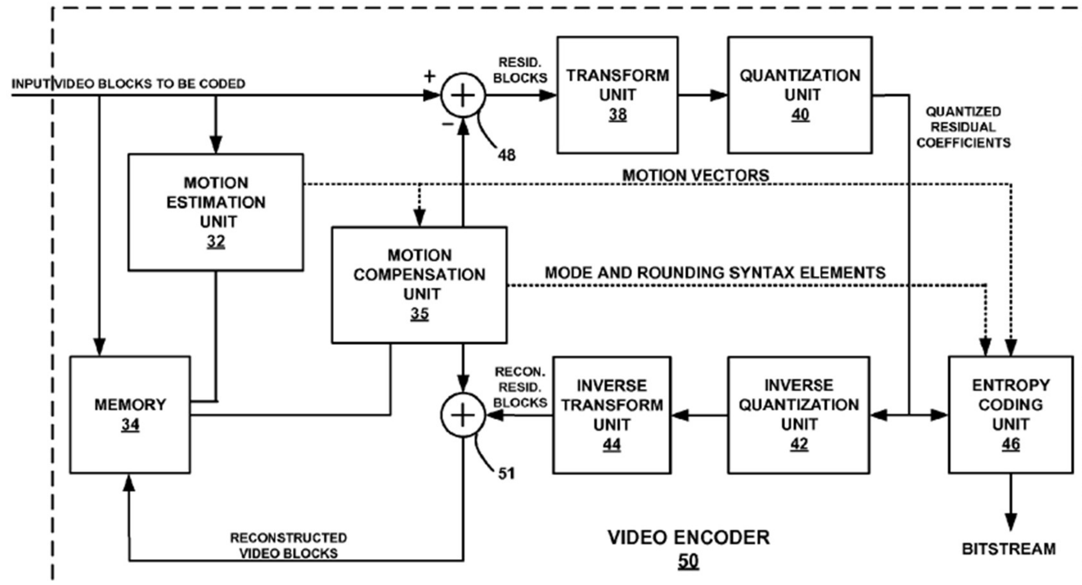
| ASSERTED CLAIMS | PRIOR ART REFERENCE:<br>KARCZEWICZ I |
|---|---|
| 1[preamble] A method for encoding a block of pixels, the method comprising: | Karczewicz I, as evidenced by the example citations below, discloses a method for encoding a block of pixels.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0002:<br><br>The disclosure relates to video encoding and, more particularly, video encoding techniques that use bi-directional prediction.<br><br>Karczewicz I, ¶0008: |

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: KARCZEWICZ I |
|---|---|
| | This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block may be predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, techniques are described for selecting among default weighted prediction, implicit weighted prediction, and explicit weighted prediction. In this context, techniques are also described for adding offset to prediction data, e.g., using the format of explicit weighted prediction to allow for offsets to predictive data that is otherwise defined by implicit or default weighted prediction.<br><br>Karczewicz I, ¶0009:<br><br>In one example, this disclosure describes a method of encoding video data, the method comprising selecting between default weighted prediction and implicit weighted prediction, upon selecting default weighted prediction, selecting between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the default weighted prediction, upon selecting implicit weighted prediction, selecting between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the implicit weighted prediction, and encoding the video data using weighed prediction data defined by the selections.<br><br>Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, Fig. 1:<br><br><br><br>FIG. 1<br><br>Karczewicz I, ¶0030: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | In the example of FIG. 1, source device 12 may include a video source 20, video encoder 22, a modulator/demodulator (modem) 23 and a transmitter 24. Destination device 16 may include a receiver 26, a modem 27, a video decoder 28, and a display device 30. In accordance with this disclosure, video encoder 22 of source device 12 may be configured to apply non-zero rounding and prediction mode selection techniques as part of a video encoding process. Video decoder 28 may receive one or more syntax elements indicating the selection and indicating whether non-zero rounding was used. Accordingly, video decoder 28 may perform the proper weighted prediction signaled in the received bistream.<br><br>Karczewicz I, ¶0032:<br><br>Video source 20 of source device 12 may include a video capture device, such as a video camera, a video archive containing previously captured video, or a video feed from a video content provider. As a further alternative, video source 20 may generate computer graphics-based data as the source video, or a combination of live video, archived video, and computer-generated video. In some cases, if video source 20 is a video camera, source device 12 and destination device 16 may form so-called camera phones or video phones. In each case, the captured, pre-captured or computer-generated video may be encoded by video encoder 22. The encoded video information may then be modulated by modem 23 according to a communication standard, e.g., such as code division multiple access (CDMA) or another communication standard, and transmitted to destination device 16 via transmitter 24. Modem 23 may include various mixers, filters, amplifiers or other components designed for signal modulation. Transmitter 24 may include circuits designed for transmitting data, including amplifiers, filters, and one or more antennas.<br><br>Karczewicz I, Fig. 2: |

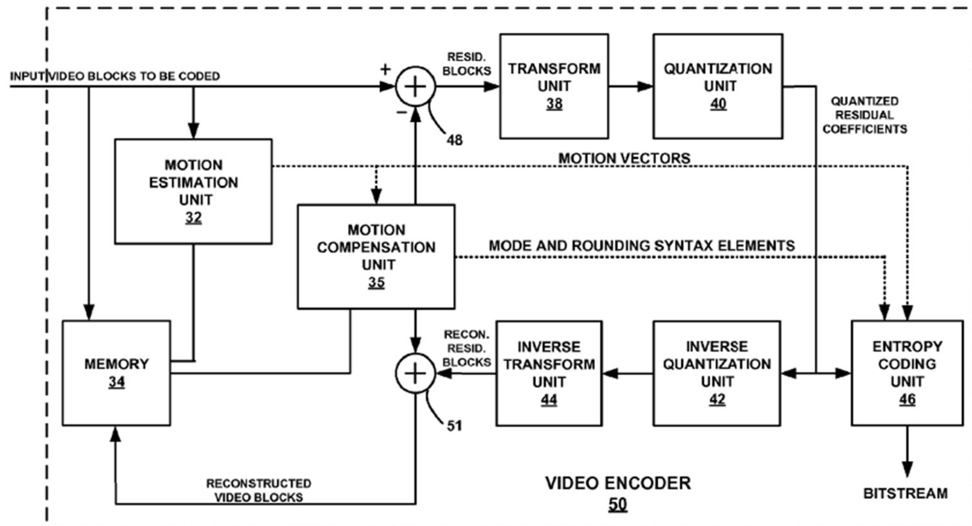
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>**FIG. 2**<br><br>Karczewicz I, ¶0039:<br><br>A video sequence typically includes a series of video frames. Video encoder 22 and video decoder 28 may operate on video blocks within individual video frames in order to encode and decode the video data. The video blocks may have fixed or varying sizes, and may differ in size according to a specified coding standard. Each video frame may include a series of slices or other independently decodable units. Each slice may include a series of macroblocks, which may be arranged into sub-blocks. As an example, the ITU-T H.264 |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | standard supports intra prediction in various block sizes, such as 16 by 16, 8 by 8, or 4 by 4 for luma components, and 8×8 for chroma components, as well as inter prediction in various block sizes, such as 16 by 16, 16 by 8, 8 by 16, 8 by 8, 8 by 4, 4 by 8 and 4 by 4 for luma components and corresponding scaled sizes for chroma components. Video blocks may comprise blocks of pixel data, or blocks of transformation coefficients, e.g., following a transformation process such as discrete cosine transform or a conceptually similar transformation process.<br><br>Karczewicz I, ¶0040:<br><br>Smaller video blocks can provide better resolution, and may be used for locations of a video frame that include high levels of detail. In general, macroblocks and the various sub-blocks may be considered to be video blocks. In addition, a slice may be considered to be a series of video blocks, such as macroblocks and/or sub-blocks. Each slice may be an independently decodable unit of a video frame. Alternatively, frames themselves may be decodable units, or other portions of a frame may be defined as decodable units. The term "coded unit" refers to any independently decodable unit of a video frame such as an entire frame, a slice of a frame, a group of pictures (GOPs), or another independently decodable unit defined according to the coding techniques used.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 1[a] determining, for a current block, a first reference block | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0005:<br><br>For inter-coding, a video encoder performs motion estimation to track the movement of matching video blocks between two or more adjacent frames or other coded units, such as slices of frames. Inter-coded frames may include predictive frames ("P-frames"), which may include blocks predicted from a previous frame, and bidirectional predictive frames ("B-frames"), which may include blocks predicted from a previous frame and a subsequent frame of a video sequence. The terms P-frames and B-frames are somewhat historic in the sense that early coding techniques limited prediction in specific directions. Newer coding formats and standards may not limit the prediction direction of P-frames or B-frames. Thus, the term "bi-directional" now refers to prediction based on two or more lists of reference data regardless of the temporal relationship of such reference data relative to the data being coded.<br><br>Karczewicz I, ¶0006:<br><br>Consistent with newer video standards such as ITU H.264, for example, bi-directional prediction may be based on two different lists which do not necessarily need to have data that resides temporally before and after the current video block. In other words, B-video blocks may be predicted from two lists of data, which may correspond to data from two previous frames, two subsequent frames, or one previous frame and one subsequent frame. In contrast, P-video blocks are predicted based on one list, i.e., one data structure, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include only I-video blocks.<br><br>Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, ¶0042:<br><br>The techniques of this disclosure are specifically applicable to weighted bi-directional prediction. As mentioned above, bi-directional prediction is prediction of so-called "B-video blocks" based on two different lists of data. B-video blocks may be predicted from two lists of data from two previous frames, two lists of data from subsequent frames, or one list of data from a previous frame and one from a subsequent frame. In contrast, P-video blocks are predicted based on one list, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | only I-video blocks.<br><br>Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units.<br><br>Karczewicz I, Fig. 2: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 2<br>Karczewicz I, ¶0053:<br>During the encoding process, video encoder 50 receives a video block to be coded, and motion estimation unit 32 and motion compensation unit 35 perform inter-predictive coding. Motion estimation unit 32 and motion compensation unit 35 may be highly integrated, but are illustrated separately for conceptual purposes. Motion estimation is typically considered the process of generating motion vectors, which estimate motion for video blocks. A motion vector, for example, may indicate the displacement of a predictive block within a predictive frame (or other coded unit) relative to the current block being coded within the current frame (or other coded unit). Motion compensation is typically considered the process of fetching or generating the predictive block based on the motion vector determined by motion estimation. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Again, motion estimation unit 32 and motion compensation unit 35 may be functionally integrated. For demonstrative purposes, the techniques described in this disclosure are described as being performed by motion compensation unit 35.<br><br>Karczewicz I, ¶0054:<br><br>Motion estimation unit 32 selects the appropriate motion vector for the video block to be coded by comparing the video block to video blocks of one or more predictive coded units (e.g., a previous and/or future frame in terms of time or temporally). Motion estimation unit 32 may, as an example, select a motion vector for a B-frame in a number of ways. In one way, motion estimation unit 32 may select a previous or future frame from a first set of frames (referred to as list 0) and determine a motion vector using only this previous or future frame from list 0. Alternatively, motion estimation unit 32 may select a previous or future frame from a second set of frames (referred to as list 1) and determine a motion vector using only this previous or future frame from list 1. In yet another way, motion estimation unit 32 may select a first frame from list 0 and a second frame from list 1 and select one or more motion vectors from the first frame of list 0 and the second frame of list 1. This third form of prediction may be referred to as bi-predictive motion estimation. Techniques of this disclosure may be implemented so as to efficiently select a motion-compensated bi-prediction mode. The selected motion vector for any given list may point to a predictive video block that is most similar to the video block being coded, e.g., as defined by a metric such as sum of absolute difference (SAD) or sum of squared difference (SSD) of pixel values of the predictive block relative to pixel values of the block being coded.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *references identified in Exhibit E-9* |
| 1[b] using said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br><br>Karczewicz I, ¶0024:<br><br>Weighted prediction refers to bi-directional prediction in which weights can be assigned to two or more different sets of predictive data. Default weighted prediction refers to weighted prediction in which the weight factors associated with two or more different lists are pre-defined by some default setting. Default weighted prediction, in some cases, may assign equal weightings to each of the lists. Implicit weighted prediction refers to weighted prediction in which the weight factors associated with two or more different lists are defined based on some implicit factors associated with the data. For example, implicit weight factors may be defined by the temporal positions of the data in the two different lists relative to the data being predictively coded. If data in list 0 is temporally closer to the data being predictively coded than data in list 1, then the data is list 0 may be assigned a greater implicit weigh factor in implicit weighted prediction.<br><br>Karczewicz I, ¶0044:<br><br>For different types of weighted bi-directional prediction in accordance with ITU-T H.264, video encoder 22 and video decoder 28 may generally support three different types of prediction modes. A first prediction mode, referred to as "default weighted prediction," refers to weighted prediction in which the weight factors associated with two or more different lists are pre-defined by some default setting. Default weighted prediction, in some cases, may assign equal weightings to each of the lists. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0048:<br><br>In accordance with another aspect of this disclosure, video encoder 22 may select among default weighted prediction, implicit weighted prediction, and explicit weighted prediction. In this aspect, rather than consider each of these possibilities relative to one another, encoder 22 may be programmed to select between default weighted prediction and implicit weighted prediction. The selection, then, may be compared to explicit weighted prediction. Specifically, encoder 22 may perform explicit weighted prediction to calculate explicit weight factors, but encoder 22 may also compare the computed explicit weighted prediction to explicit weighted prediction having weight factors that correspond to those defined by either default weighted prediction or implicit weighted prediction.<br><br>Karczewicz I, ¶0055:<br><br>According to the ITU-T H.264/AVC standard, three motion-compensated bi-predictive algorithms or modes may be used to predict a B-frame or portions thereof, such as video blocks, macroblocks or any other discreet and/or contiguous portion of a B-frame. A first motion-compensated bi-predictive algorithm or mode, which is commonly referred to as default weighted prediction, may involve applying default weights to each identified video block of the first frame of list 0 and the second frame of list 1. The default weights may be programmed according to the standard, and are often selected to be equal for default weighted prediction. The weighted blocks of the first and second frames are then added together and divided by the total number of frames used to predict the B-frame, e.g., two in this instance. Often, this division is accomplished by adding 1 to the addition of the weighted blocks of the first and second frames and then shifting the result to the right by one bit. The addition of 1 is a rounding adjustment. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0060:<br><br>Default weighted prediction may be defined by the following equations for unidirectional prediction and bidirectional prediction, respectively.<br><br>Unidirectional prediction: pred(i,j)=pred0(i,j)<br><br>Bidirectional prediction: pred(i,j)=(pred0(i,j)+pred1(i,j)+1)>>1<br>where pred0(i,j) and pred1(i,j) are prediction data from list 0 and list 1.<br><br>Karczewicz I, ¶0058:<br><br>B pictures use two lists of previously-coded reference pictures, list 0 and list 1. These two lists can each contain past and/or future coded pictures in temporal order. Blocks in a B picture may be predicted in one of several ways: motion-compensated prediction from a list 0 reference picture, motion-compensated prediction from a list 1 reference picture, or motion-compensated prediction from the combination of both list 0 and list 1 reference pictures. To get the combination of both list 0 and list 1 reference pictures, two motion compensated reference areas are obtained from list 0 and list 1 reference picture respectively. Their combination will be used to predict the current block.<br><br>Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units.<br><br>Karczewicz I, ¶0041:<br><br>Following inter-based predictive encoding (which includes interpolation and the techniques of this disclosure to efficiently select a prediction algorithm or mode by which to predict a coded unit), and following any transforms (such as the 4×4 or 8×8 integer transform used in H.264/AVC or a discrete cosine transform or DCT), quantization may be performed. Quantization generally refers to a process in which coefficients are quantized to possibly reduce the amount of data used to represent the coefficients. The quantization process may reduce the bit depth associated with some or all of the coefficients. For example, a 16-bit value may be rounded down to a 15-bit value during quantization. Following quantization, entropy coding may be performed, e.g., according to content adaptive variable length coding (CAVLC), context adaptive binary arithmetic coding (CABAC), or another entropy coding methodology.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 1[c] using said second reference block to obtain a second prediction, said second prediction having the second | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| precision; | Karczewicz I, ¶0024:<br><br>Weighted prediction refers to bi-directional prediction in which weights can be assigned to two or more different sets of predictive data. Default weighted prediction refers to weighted prediction in which the weight factors associated with two or more different lists are pre-defined by some default setting. Default weighted prediction, in some cases, may assign equal weightings to each of the lists. Implicit weighted prediction refers to weighted prediction in which the weight factors associated with two or more different lists are defined based on some implicit factors associated with the data. For example, implicit weight factors may be defined by the temporal positions of the data in the two different lists relative to the data being predictively coded. If data in list 0 is temporally closer to the data being predictively coded than data in list 1, then the data is list 0 may be assigned a greater implicit weigh factor in implicit weighted prediction.<br><br>Karczewicz I, ¶0044:<br><br>For different types of weighted bi-directional prediction in accordance with ITU-T H.264, video encoder 22 and video decoder 28 may generally support three different types of prediction modes. A first prediction mode, referred to as "default weighted prediction," refers to weighted prediction in which the weight factors associated with two or more different lists are pre-defined by some default setting. Default weighted prediction, in some cases, may assign equal weightings to each of the lists.<br><br>Karczewicz I, ¶0048:<br><br>In accordance with another aspect of this disclosure, video encoder 22 may select among default weighted prediction, implicit weighted prediction, and explicit weighted prediction. In this aspect, rather than consider each of these possibilities relative to one another, encoder 22 |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | may be programmed to select between default weighted prediction and implicit weighted prediction. The selection, then, may be compared to explicit weighted prediction. Specifically, encoder 22 may perform explicit weighted prediction to calculate explicit weight factors, but encoder 22 may also compare the computed explicit weighted prediction to explicit weighted prediction having weight factors that correspond to those defined by either default weighted prediction or implicit weighted prediction.<br><br>Karczewicz I, ¶0055:<br><br>According to the ITU-T H.264/AVC standard, three motion-compensated bi-predictive algorithms or modes may be used to predict a B-frame or portions thereof, such as video blocks, macroblocks or any other discreet and/or contiguous portion of a B-frame. A first motion-compensated bi-predictive algorithm or mode, which is commonly referred to as default weighted prediction, may involve applying default weights to each identified video block of the first frame of list 0 and the second frame of list 1. The default weights may be programmed according to the standard, and are often selected to be equal for default weighted prediction. The weighted blocks of the first and second frames are then added together and divided by the total number of frames used to predict the B-frame, e.g., two in this instance. Often, this division is accomplished by adding 1 to the addition of the weighted blocks of the first and second frames and then shifting the result to the right by one bit. The addition of 1 is a rounding adjustment.<br><br>Karczewicz I, ¶0060:<br><br>Default weighted prediction may be defined by the following equations for unidirectional prediction and bidirectional prediction, respectively.<br><br>Unidirectional prediction: pred(i,j)=pred0(i,j) |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Bidirectional prediction: pred(i,j)=(pred0(i,j)+pred1(i,j)+1)>>1<br>where pred0(i,j) and pred1(i,j) are prediction data from list 0 and list 1.<br><br>Karczewicz I, ¶0058:<br><br>B pictures use two lists of previously-coded reference pictures, list 0 and list 1. These two lists can each contain past and/or future coded pictures in temporal order. Blocks in a B picture may be predicted in one of several ways: motion-compensated prediction from a list 0 reference picture, motion-compensated prediction from a list 1 reference picture, or motion-compensated prediction from the combination of both list 0 and list 1 reference pictures. To get the combination of both list 0 and list 1 reference pictures, two motion compensated reference areas are obtained from list 0 and list 1 reference picture respectively. Their combination will be used to predict the current block.<br><br>Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0041:<br><br>Following inter-based predictive encoding (which includes interpolation and the techniques of this disclosure to efficiently select a prediction algorithm or mode by which to predict a coded unit), and following any transforms (such as the 4×4 or 8×8 integer transform used in H.264/AVC or a discrete cosine transform or DCT), quantization may be performed. Quantization generally refers to a process in which coefficients are quantized to possibly reduce the amount of data used to represent the coefficients. The quantization process may reduce the bit depth associated with some or all of the coefficients. For example, a 16-bit value may be rounded down to a 15-bit value during quantization. Following quantization, entropy coding may be performed, e.g., according to content adaptive variable length coding (CAVLC), context adaptive binary arithmetic coding (CABAC), or another entropy coding methodology.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 1[d] obtaining a combined prediction based at least partly upon said first prediction and said second prediction; | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined prediction based at least partly upon the first prediction and the second prediction.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0060:<br><br>Default weighted prediction may be defined by the following equations for unidirectional prediction and bidirectional prediction, respectively. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Unidirectional prediction: pred(i,j)=pred0(i,j)<br><br>Bidirectional prediction: pred(i,j)=(pred0(i,j)+pred1(i,j)+1)>>1<br>where pred0(i,j) and pred1(i,j) are prediction data from list 0 and list 1.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 1[e] decreasing a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0060:<br><br>Default weighted prediction may be defined by the following equations for unidirectional prediction and bidirectional prediction, respectively.<br><br>Unidirectional prediction: pred(i,j)=pred0(i,j)<br><br>Bidirectional prediction: pred(i,j)=(pred0(i,j)+pred1(i,j)+1)>>1<br>where pred0(i,j) and pred1(i,j) are prediction data from list 0 and list 1.<br><br>Karczewicz I, ¶0055: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | According to the ITU-T H.264/AVC standard, three motion-compensated bi-predictive algorithms or modes may be used to predict a B-frame or portions thereof, such as video blocks, macroblocks or any other discreet and/or contiguous portion of a B-frame. A first motion-compensated bi-predictive algorithm or mode, which is commonly referred to as default weighted prediction, may involve applying default weights to each identified video block of the first frame of list 0 and the second frame of list 1. The default weights may be programmed according to the standard, and are often selected to be equal for default weighted prediction. The weighted blocks of the first and second frames are then added together and divided by the total number of frames used to predict the B-frame, e.g., two in this instance. Often, this division is accomplished by adding 1 to the addition of the weighted blocks of the first and second frames and then shifting the result to the right by one bit. The addition of 1 is a rounding adjustment.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 1[f] encoding residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined prediction and the block of pixels. | Karczewicz I, as evidenced by the example citations below, discloses encoding residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined prediction and the block of pixels.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units.<br><br>Karczewicz I, ¶0073:<br><br>Once the desired prediction data is identified by motion compensation unit 35, as described herein, video encoder 50 forms a residual video block by subtracting the prediction data from the original video block being coded. Adder 48 represents the component or components that perform this subtraction operation. Transform unit 38 applies a transform, such as a discrete cosine transform (DCT) or a conceptually similar transform, to the residual block, producing a video block comprising residual transform block coefficients. Transform unit 38, for example, may perform other transforms, such as those defined by the H.264 standard, which are conceptually similar to DCT. Wavelet transforms, integer transforms, sub-band transforms or other types of transforms could also be used. In any case, transform unit 38 applies the transform to the residual block, producing a block of residual transform coefficients. The transform may convert the residual information from a pixel domain to a frequency domain.<br><br>Karczewicz I, ¶0074:<br><br>Quantization unit 40 quantizes the residual transform coefficients to further reduce bit rate. The quantization process may reduce the bit depth associated with some or all of the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | coefficients. For example, a 9-bit value may be rounded down to an 8-bit value during quantization. In addition, quantization unit 40 may also quantize the different offsets for cases where offset is used.<br><br>Karczewicz I, ¶0075:<br><br>Following quantization, entropy coding unit 46 entropy codes the quantized transform coefficients. For example, entropy coding unit 46 may perform content adaptive variable length coding (CAVLC), context adaptive binary arithmetic coding (CABAC), or another entropy coding methodology. Following the entropy coding by entropy coding unit 46, the encoded video may be transmitted to another device or archived for later transmission or retrieval. The coded bitstream may include entropy coded residual blocks, motion vectors for such blocks, and other syntax such as the syntax described herein.<br><br>Karczewicz I, ¶0084:<br><br>Video decoder 70 may receive encoded video data, and one or more syntax elements that indicate whether a rounding adjustment was used to encode the encoded video data. MC unit 86 of prediction unit 75 may generate weighted prediction data that depends on two or more lists of data, as described herein. In accordance with this disclosure, the weighted prediction data does not include the rounding adjustment if the one or more syntax elements indicate that the rounding adjustment was not used to encode the encoded video data. Video decoder 70 can decode the video data using the weighted prediction data, e.g., by invoking adder 79 to add the weighted prediction data (e.g., a prediction block) to residual data (e.g., a residual block).<br><br>Karczewicz I, ¶0089: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE:<br>KARCZEWICZ I |
|---|---|
| | Prediction unit 75 invokes motion compensation unit 86 for block based predictive decoding. In doing so, motion compensation unit 86 generates weighted prediction data based on the syntax (113). Accordingly, if the one or more syntax elements indicate that a rounding adjustment was used, then motion compensation unit 86 generates weighted prediction data that includes the rounding adjustment. However, if the one or more syntax elements indicate that a rounding adjustment was not used, then motion compensation unit 86 generates weighted prediction data that lacks the rounding adjustment. Video decoder 70 can then decode the video data using the weighted prediction data (114). In particular, video decoder 70 may invoke adder 79 to combine weighted prediction data (e.g., a prediction block) with residual video data (e.g., a residual block) in order to generate a reconstruction of the video data (e.g., a reconstructed video block).<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 2. The method according to claim 1, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel values of the first reference block.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0041:<br><br>Following inter-based predictive encoding (which includes interpolation and the techniques of this disclosure to efficiently select a prediction algorithm or mode by which to predict a coded unit), and following any transforms (such as the 4×4 or 8×8 integer transform used in |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | H.264/AVC or a discrete cosine transform or DCT), quantization may be performed. Quantization generally refers to a process in which coefficients are quantized to possibly reduce the amount of data used to represent the coefficients. The quantization process may reduce the bit depth associated with some or all of the coefficients. For example, a 16-bit value may be rounded down to a 15-bit value during quantization. Following quantization, entropy coding may be performed, e.g., according to content adaptive variable length coding (CAVLC), context adaptive binary arithmetic coding (CABAC), or another entropy coding methodology.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 3. The method according to claim 2, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter using values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 4. The method according to claim 2, wherein in an instance in which said second | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| motion vector points to an integer sample, said second prediction is obtained by shifting values of said second reference block to the left. | of the second reference block to the left.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 5. The method according to claim 1, wherein said decreasing said precision of said combined prediction by shifting bits of the combined prediction to the right, further comprises: inserting a rounding offset to the combined prediction before said decreasing. | Karczewicz I, as evidenced by the example citations below, discloses that the decreasing the precision of the combined prediction by shifting bits of the combined prediction to the right, further comprises inserting a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0060:<br><br>Default weighted prediction may be defined by the following equations for unidirectional prediction and bidirectional prediction, respectively.<br>Unidirectional prediction: pred(i,j)=pred0(i,j)<br>Bidirectional prediction: pred(i,j)=(pred0(i,j)+pred1(i,j)+1)>>1<br>where pred0(i,j) and pred1(i,j) are prediction data from list 0 and list 1.<br><br>Karczewicz I, ¶0055:<br><br>According to the ITU-T H.264/AVC standard, three motion-compensated bi-predictive algorithms or modes may be used to predict a B-frame or portions thereof, such as video blocks, macroblocks or any other discreet and/or contiguous portion of a B-frame. A first motion-compensated bi-predictive algorithm or mode, which is commonly referred to as |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | default weighted prediction, may involve applying default weights to each identified video block of the first frame of list 0 and the second frame of list 1. The default weights may be programmed according to the standard, and are often selected to be equal for default weighted prediction. The weighted blocks of the first and second frames are then added together and divided by the total number of frames used to predict the B-frame, e.g., two in this instance. Often, this division is accomplished by adding 1 to the addition of the weighted blocks of the first and second frames and then shifting the result to the right by one bit. The addition of 1 is a rounding adjustment.<br><br>Karczewicz I, ¶0063:<br><br>Conventionally, rounding adjustments are always used in bidirectional prediction. According to the equations above, a rounding adjustment of 1 is used in the default weighted prediction prior to a right shift by one, and a rounding adjustment of 32 is used in the implicit weighted prediction prior to a right shift by six. Generally, a rounding adjustment of 2r−1 is commonly used prior to a right shift by r, where r represents a positive integer.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 6. The method according to claim 1, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of the first prediction and values of the second prediction. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| second precision indicates the number of bits needed to represent values of said first prediction and values of said second prediction. | *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 7[preamble] An apparatus for encoding a block of pixels, the apparatus comprising: | Karczewicz I, as evidenced by the example citations below, discloses an apparatus for encoding a block of pixels.<br><br>*See, e.g.*,<br><br>*See limitation 1[preamble].*<br><br>Karczewicz I, ¶0003:<br><br>> Digital multimedia capabilities can be incorporated into a wide range of devices, including digital televisions, digital direct broadcast systems, wireless communication devices, wireless broadcast systems, personal digital assistants (PDAs), laptop or desktop computers, digital cameras, digital recording devices, video gaming devices, video game consoles, cellular or satellite radio telephones, digital media players, and the like. Digital multimedia devices may implement video coding techniques, such as MPEG-2, ITU-H.263, MPEG-4, or ITU-H.264/MPEG-4 Part 10, Advanced Video Coding (AVC), to transmit and receive or store and retrieve digital video data more efficiently. Video encoding techniques may perform video compression via spatial and temporal prediction to reduce or remove redundancy inherent in video sequences.<br><br>Karczewicz I, ¶0010: |

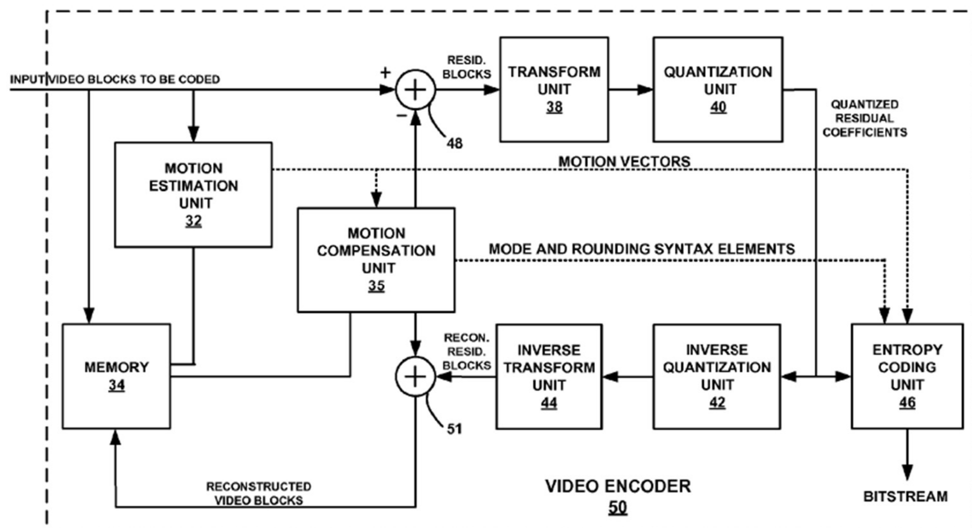
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | In another example, this disclosure describes a video encoder apparatus that encodes video data, the video encoder apparatus comprising a memory that stores the video data, and a motion compensation unit that selects between default weighted prediction and implicit weighted prediction, upon selecting default weighted prediction, selects between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the default weighted prediction, upon selecting implicit weighted prediction, selects between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the implicit weighted prediction, wherein the video encoder apparatus encodes the video data using weighed prediction data defined by the selections.<br><br>Karczewicz I, ¶0012:<br><br>The techniques described in this disclosure may be implemented in hardware, software, firmware, or any combination thereof If implemented in software, the software may be executed in one or more processors, such as a microprocessor, application specific integrated circuit (ASIC), field programmable gate array (FPGA), or digital signal processor (DSP). The software that executes the techniques may be initially stored in a computer-readable medium and loaded and executed in the processor.<br><br>Karczewicz I, ¶0038:<br><br>Video encoder 22 and video decoder 28 each may be implemented as one or more microprocessors, digital signal processors (DSPs), application specific integrated circuits (ASICs), field programmable gate arrays (FPGAs), discrete logic, software, hardware, firmware or any combinations thereof. Each of video encoder 22 and video decoder 28 may be included in one or more encoders or decoders, either of which may be integrated as part of |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | a combined encoder/decoder (CODEC) in a respective mobile device, subscriber device, broadcast device, server, or the like.<br><br>Karczewicz I, ¶0098:<br><br>The techniques of this disclosure may be implemented in a wide variety of devices or apparatuses, including a wireless handset, and integrated circuit (IC) or a set of ICs (i.e., a chip set). Any components, modules or units have been described provided to emphasize functional aspects and does not necessarily require realization by different hardware units. The techniques described herein may also be implemented in hardware, software, firmware, or any combination thereof. Any features described as modules, units or components may be implemented together in an integrated logic device or separately as discrete but interoperable logic devices. In some cases, various features may be implemented as an integrated circuit device, such as an integrated circuit chip or chipset.<br><br>Karczewicz I, ¶0099:<br><br>If implemented in software, the techniques may be realized at least in part by a computer-readable medium comprising instructions that, when executed in a processor, performs one or more of the methods described above. The computer-readable medium may comprise a computer-readable storage medium and may form part of a computer program product, which may include packaging materials. The computer-readable storage medium may comprise random access memory (RAM) such as synchronous dynamic random access memory (SDRAM), read-only memory (ROM), non-volatile random access memory (NVRAM), electrically erasable programmable read-only memory (EEPROM), FLASH memory, magnetic or optical data storage media, and the like. The techniques additionally, or alternatively, may be realized at least in part by a computer-readable communication medium that carries or communicates code in the form of instructions or data structures and that can |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | be accessed, read, and/or executed by a computer.<br><br>Karczewicz I, ¶0100:<br><br>The code or instructions may be executed by one or more processors, such as one or more digital signal processors (DSPs), general purpose microprocessors, an application specific integrated circuits (ASICs), field programmable logic arrays (FPGAs), or other equivalent integrated or discrete logic circuitry. Accordingly, the term "processor," as used herein may refer to any of the foregoing structure or any other structure suitable for implementation of the techniques described herein. In addition, in some aspects, the functionality described herein may be provided within dedicated software modules or hardware modules configured for encoding and decoding, or incorporated in a combined video codec. Also, the techniques could be fully implemented in one or more circuits or logic elements.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 7[a] at least one processor and at least one memory including computer program code, the at least one memory and computer program code configured to, with the at least one processor, cause the apparatus to: | Karczewicz I, as evidenced by the example citations below, discloses at least one processor and at least one memory including computer program code.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0005:<br><br>For inter-coding, a video encoder performs motion estimation to track the movement of matching video blocks between two or more adjacent frames or other coded units, such as |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | slices of frames. Inter-coded frames may include predictive frames ("P-frames"), which may include blocks predicted from a previous frame, and bidirectional predictive frames ("B-frames"), which may include blocks predicted from a previous frame and a subsequent frame of a video sequence. The terms P-frames and B-frames are somewhat historic in the sense that early coding techniques limited prediction in specific directions. Newer coding formats and standards may not limit the prediction direction of P-frames or B-frames. Thus, the term "bi-directional" now refers to prediction based on two or more lists of reference data regardless of the temporal relationship of such reference data relative to the data being coded.<br><br>Karczewicz I, ¶0006:<br><br>Consistent with newer video standards such as ITU H.264, for example, bi-directional prediction may be based on two different lists which do not necessarily need to have data that resides temporally before and after the current video block. In other words, B-video blocks may be predicted from two lists of data, which may correspond to data from two previous frames, two subsequent frames, or one previous frame and one subsequent frame. In contrast, P-video blocks are predicted based on one list, i.e., one data structure, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include only I-video blocks.<br><br>Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, ¶0042:<br><br>The techniques of this disclosure are specifically applicable to weighted bi-directional prediction. As mentioned above, bi-directional prediction is prediction of so-called “B-video blocks” based on two different lists of data. B-video blocks may be predicted from two lists of data from two previous frames, two lists of data from subsequent frames, or one list of data from a previous frame and one from a subsequent frame. In contrast, P-video blocks are predicted based on one list, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include only I-video blocks.<br><br>Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units.<br><br>Karczewicz I, Fig. 2:<br><br><br><br>FIG. 2<br><br>Karczewicz I, ¶0053:<br><br>During the encoding process, video encoder 50 receives a video block to be coded, and motion estimation unit 32 and motion compensation unit 35 perform inter-predictive coding. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Motion estimation unit 32 and motion compensation unit 35 may be highly integrated, but are illustrated separately for conceptual purposes. Motion estimation is typically considered the process of generating motion vectors, which estimate motion for video blocks. A motion vector, for example, may indicate the displacement of a predictive block within a predictive frame (or other coded unit) relative to the current block being coded within the current frame (or other coded unit). Motion compensation is typically considered the process of fetching or generating the predictive block based on the motion vector determined by motion estimation. Again, motion estimation unit 32 and motion compensation unit 35 may be functionally integrated. For demonstrative purposes, the techniques described in this disclosure are described as being performed by motion compensation unit 35. <br><br> Karczewicz I, ¶0054: <br><br> Motion estimation unit 32 selects the appropriate motion vector for the video block to be coded by comparing the video block to video blocks of one or more predictive coded units (e.g., a previous and/or future frame in terms of time or temporally). Motion estimation unit 32 may, as an example, select a motion vector for a B-frame in a number of ways. In one way, motion estimation unit 32 may select a previous or future frame from a first set of frames (referred to as list 0) and determine a motion vector using only this previous or future frame from list 0. Alternatively, motion estimation unit 32 may select a previous or future frame from a second set of frames (referred to as list 1) and determine a motion vector using only this previous or future frame from list 1. In yet another way, motion estimation unit 32 may select a first frame from list 0 and a second frame from list 1 and select one or more motion vectors from the first frame of list 0 and the second frame of list 1. This third form of prediction may be referred to as bi-predictive motion estimation. Techniques of this disclosure may be implemented so as to efficiently select a motion-compensated bi-prediction mode. The selected motion vector for any given list may point to a predictive video block that is most similar to the video block being coded, e.g., as defined by a metric such as sum of |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | absolute difference (SAD) or sum of squared difference (SSD) of pixel values of the predictive block relative to pixel values of the block being coded.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 7[b] determine, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision.<br><br>*See, e.g.*,<br><br>*See limitation 1[a].* |
| 7[c] use said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br><br>*See, e.g.*,<br><br>*See limitation 1[b].* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| 7[d] use said second reference block to obtain a second prediction, said second prediction having the second precision; | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision.<br><br>*See, e.g.*,<br><br>*See limitation 1[c]*. |
| 7[e] obtain a combined prediction based at least partly upon said first prediction and said second prediction; | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined prediction based at least partly upon the first prediction and the second prediction.<br><br>*See, e.g.*,<br><br>*See limitation 1[d]*. |
| 7[f] decrease a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br><br>*See, e.g.*,<br><br>*See limitation 1[e]*. |
| 7[g] encode residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined | Karczewicz I, as evidenced by the example citations below, discloses encoding residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined prediction and the block of pixels.<br><br>*See, e.g.*, |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| prediction and the block of pixels. | *See limitation 1[f].* |
| 8. The apparatus according to claim 7, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel values of the first reference block.<br><br>*See, e.g.*,<br><br>*See claim 2.* |
| 9. The apparatus according to claim 8, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter using values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br><br>*See, e.g.*,<br><br>*See claim 3.* |
| 10. The apparatus according | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| to claim 8, wherein in an instance in which said second motion vector points to an integer sample, said second prediction is obtained by shifting values of said second reference block to the left. | of the second reference block to the left.<br><br>*See, e.g.*,<br><br>*See claim 4.* |
| 11. The apparatus according to claim 7, wherein the at least one memory and computer code are configured to cause the apparatus to decrease said precision of said combined prediction by shifting bits of the combined prediction to the right, by: inserting a rounding offset to the combined prediction before said decreasing. | Karczewicz I, as evidenced by the example citations below, discloses that the at least one memory and computer code are configured to cause the apparatus to decrease the precision of the combined prediction by shifting bits of the combined prediction to the right, by inserting a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.*,<br><br>*See claim 5.*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 12. The apparatus according to claim 7, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of the first prediction and values of the second prediction. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| number of bits needed to represent values of said first prediction and values of said second prediction. | *See, e.g.*,<br><br>*See claim 6.* |
| 13[preamble] A computer program product for encoding a block of pixels, the computer program product comprising at least one non-transitory computer readable storage medium having computer executable program code portions stored therein, the computer executable program code portions comprising program code instructions configured to: | Karczewicz I, as evidenced by the example citations below, discloses a computer program product for encoding a block of pixels, the computer program product comprising at least one non-transitory computer readable storage medium having computer executable program code portions stored therein.<br><br>*See, e.g.*,<br><br>*See limitations 1[preamble], 7[preamble], and 7[a].*<br><br>*See, e.g.*, Karczewicz I, ¶0013:<br><br>Accordingly, this disclosure also contemplates a computer-readable storage medium comprising instructions that when executed cause a processor to select between default weighted prediction and implicit weighted prediction, upon selecting default weighted prediction, select between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the default weighted prediction, upon selecting implicit weighted prediction, select between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the implicit weighted prediction, and encode the video data using weighed prediction data defined by the selections. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| 13[a] determine, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision.<br>*See, e.g.,*<br>*See limitation 1[a].* |
| 13[b] use said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br>*See, e.g.,*<br>*See limitation 1[b].* |
| 13[c] use said second reference block to obtain a second prediction, said second prediction having the second precision; | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision.<br>*See, e.g.,*<br>*See limitation 1[c].* |
| 13[d] obtain a combined prediction based at least partly | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined |

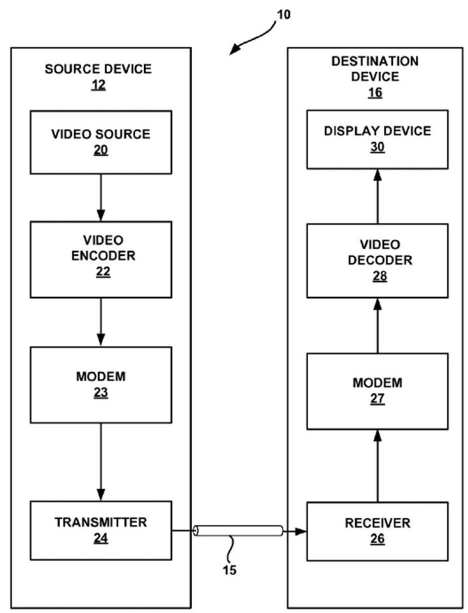
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
| --- | --- |
| upon said first prediction and said second prediction; | prediction based at least partly upon the first prediction and the second prediction.<br>*See, e.g.*,<br>*See limitation 1[d].* |
| 13[e] decrease a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br>*See, e.g.*,<br>*See limitation 1[e].* |
| 13[f] encode residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined prediction and the block of pixels. | Karczewicz I, as evidenced by the example citations below, discloses encoding residual data in a bitstream, wherein the residual data is determined based upon a difference between the combined prediction and the block of pixels.<br>*See, e.g.*,<br>*See limitation 1[f].* |
| 14. The computer program | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
| --- | --- |
| product according to claim 13, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | values of the first reference block.<br><br>*See, e.g.*,<br><br>*See claim 2.* |
| 15. The computer program product according to claim 14, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter using values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br><br>*See, e.g.*,<br><br>*See* claim 3. |
| 16. The computer program product according to claim 14, wherein in an instance in which said second motion vector points to an integer sample, said second prediction is obtained by shifting values of said second | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values of the second reference block to the left.<br><br>*See, e.g.*, |

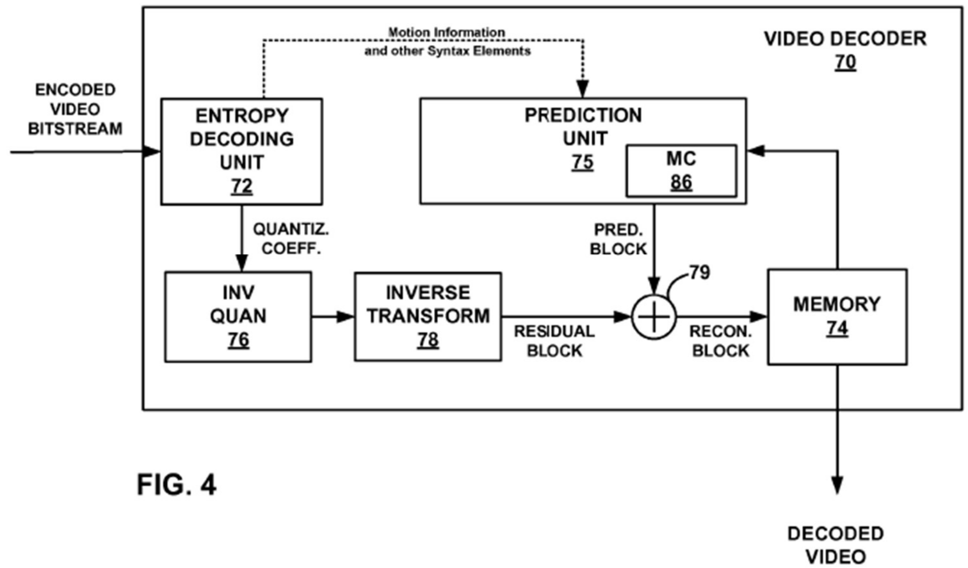
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| reference block to the left. | *See* claim 4. |
| 17. The computer program product according to claim 13, wherein the program code instructions configured to decrease said precision of said combined prediction by shifting bits of the combined prediction to the right, further comprise program code instructions configured to: insert a rounding offset to the combined prediction before said decreasing. | Karczewicz I, as evidenced by the example citations below, discloses that the program code instructions configured to decrease the precision of the combined prediction by shifting bits of the combined prediction to the right, further comprise program code instructions configured to insert a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.*,<br><br>*See claims 5 and 11.* |
| 18. The computer program product according to claim 13, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of said first prediction and values of said second prediction. | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of the first prediction and values of the second prediction.<br><br>*See, e.g.*,<br><br>*See claim 6.* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| 19[preamble] A method for decoding a block of pixels, the method comprising: | Karczewicz I, as evidenced by the example citations below, discloses a method for decoding a block of pixels.<br><br>*See, e.g.*,<br><br>Karczewicz I, ¶0008:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block may be predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, techniques are described for selecting among default weighted prediction, implicit weighted prediction, and explicit weighted prediction. In this context, techniques are also described for adding offset to prediction data, e.g., using the format of explicit weighted prediction to allow for offsets to predictive data that is otherwise defined by implicit or default weighted prediction.<br><br>Karczewicz I, ¶0009:<br><br>In one example, this disclosure describes a method of encoding video data, the method comprising selecting between default weighted prediction and implicit weighted prediction, upon selecting default weighted prediction, selecting between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the default weighted prediction, upon selecting implicit weighted prediction, selecting between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the implicit weighted prediction, and encoding the video data using weighed prediction data defined by the selections. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, Fig. 1: |

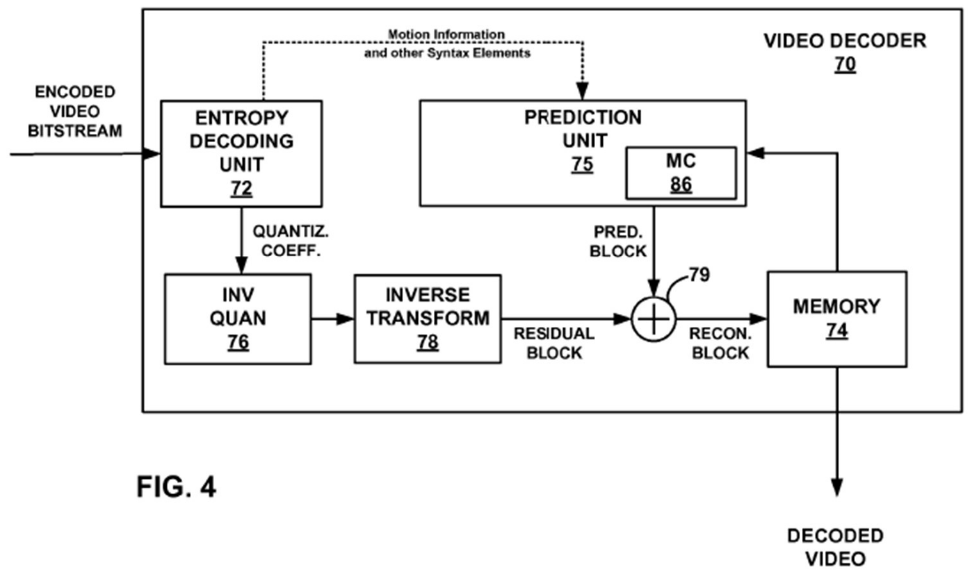
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 1<br><br>Karczewicz I, ¶0030:<br><br>In the example of FIG. 1, source device 12 may include a video source 20, video encoder 22, a modulator/demodulator (modem) 23 and a transmitter 24. Destination device 16 may include a receiver 26, a modem 27, a video decoder 28, and a display device 30. In accordance with this disclosure, video encoder 22 of source device 12 may be configured to apply non-zero rounding and prediction mode selection techniques as part of a video encoding process. Video decoder 28 may receive one or more syntax elements indicating the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | selection and indicating whether non-zero rounding was used. Accordingly, video decoder 28 may perform the proper weighted prediction signaled in the received bistream.<br><br>Karczewicz I, ¶0033:<br><br>Receiver 26 of destination device 16 receives information over channel 15, and modem 27 demodulates the information. Again, the video encoding process may implement one or more of the techniques described herein to provide non-zero rounding and prediction mode selection consistent with this disclosure. The information communicated over channel 15 may include information defined by video encoder 22, which may be used by video decoder 28 consistent with this disclosure. Display device 30 displays the decoded video data to a user, and may comprise any of a variety of display devices such as a cathode ray tube, a liquid crystal display (LCD), a plasma display, an organic light emitting diode (OLED) display, or another type of display device.<br><br>Karczewicz I, Fig. 4: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 4<br><br>Karczewicz I, ¶0039:<br><br>A video sequence typically includes a series of video frames. Video encoder 22 and video decoder 28 may operate on video blocks within individual video frames in order to encode and decode the video data. The video blocks may have fixed or varying sizes, and may differ in size according to a specified coding standard. Each video frame may include a series of slices or other independently decodable units. Each slice may include a series of macroblocks, which may be arranged into sub-blocks. As an example, the ITU-T H.264 standard supports intra prediction in various block sizes, such as 16 by 16, 8 by 8, or 4 by 4 |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | for luma components, and 8×8 for chroma components, as well as inter prediction in various block sizes, such as 16 by 16, 16 by 8, 8 by 16, 8 by 8, 8 by 4, 4 by 8 and 4 by 4 for luma components and corresponding scaled sizes for chroma components. Video blocks may comprise blocks of pixel data, or blocks of transformation coefficients, e.g., following a transformation process such as discrete cosine transform or a conceptually similar transformation process.<br><br>Karczewicz I, ¶0040:<br><br>Smaller video blocks can provide better resolution, and may be used for locations of a video frame that include high levels of detail. In general, macroblocks and the various sub-blocks may be considered to be video blocks. In addition, a slice may be considered to be a series of video blocks, such as macroblocks and/or sub-blocks. Each slice may be an independently decodable unit of a video frame. Alternatively, frames themselves may be decodable units, or other portions of a frame may be defined as decodable units. The term "coded unit" refers to any independently decodable unit of a video frame such as an entire frame, a slice of a frame, a group of pictures (GOPs), or another independently decodable unit defined according to the coding techniques used.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[a] determining, for a current block, a first reference block based on a first motion | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | second reference block have values with a first precision.<br><br>*See, e.g.,*<br><br>*See limitation 1[a].*<br><br>Karczewicz I, ¶0085:<br><br>In general, entropy decoding unit 72 receives an encoded bitsteam and entropy decodes the bitstream to generate quantized coefficients, motion information and other syntax. The motion information (e.g., motion vectors) and other syntax are forwarded to prediction unit 75 for use in generating the predictive data. Prediction unit 75 performs bidirectional prediction consistent with this disclosure, avoiding rounding adjustments in some cases, and possibly implementing default, implicit or explicit weighted prediction according to the received syntax elements. The syntax elements may identify the type of weighted prediction that to be used, coefficients and offset if explicit weighted prediction to be used, and whether rounding adjustments should be used in the decoding.<br><br>Karczewicz I, ¶0088:<br><br>FIG. 6 is a flow chart illustrating an exemplary process performed by a video decoder consistent with this disclosure. FIG. 6 will be described from the perspective of video decoder 70 of FIG. 4. As shown in FIG. 6, video decoder receives encoded video data (111), and receives one or more syntax elements that indicate whether a rounding adjustment was used to encode the video data (1 12). In particular, entropy decoding unit 72 may receive an encoded bitstream that includes the video data and the one or more syntax elements. Following entropy decoding, entropy decoding unit 72 may output the video data as quantized transform coefficients, which are inverse quantized by unit 76 and inverse |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | transformed by unit 78. Entropy decoding unit 72 may output syntax elements to prediction unit, which includes the one or more syntax elements that indicate whether a rounding adjustment was used to encode the video data, motion vectors and possibly other syntax. <br><br> Karczewicz I, Fig. 4: <br><br>   <br> FIG. 4 <br><br> *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[b] using said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br><br>*See, e.g.*,<br><br>*See limitation 1[b].*<br><br>Karczewicz I, ¶0008:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block may be predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, techniques are described for selecting among default weighted prediction, implicit weighted prediction, and explicit weighted prediction. In this context, techniques are also described for adding offset to prediction data, e.g., using the format of explicit weighted prediction to allow for offsets to predictive data that is otherwise defined by implicit or default weighted prediction.<br><br>Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, |

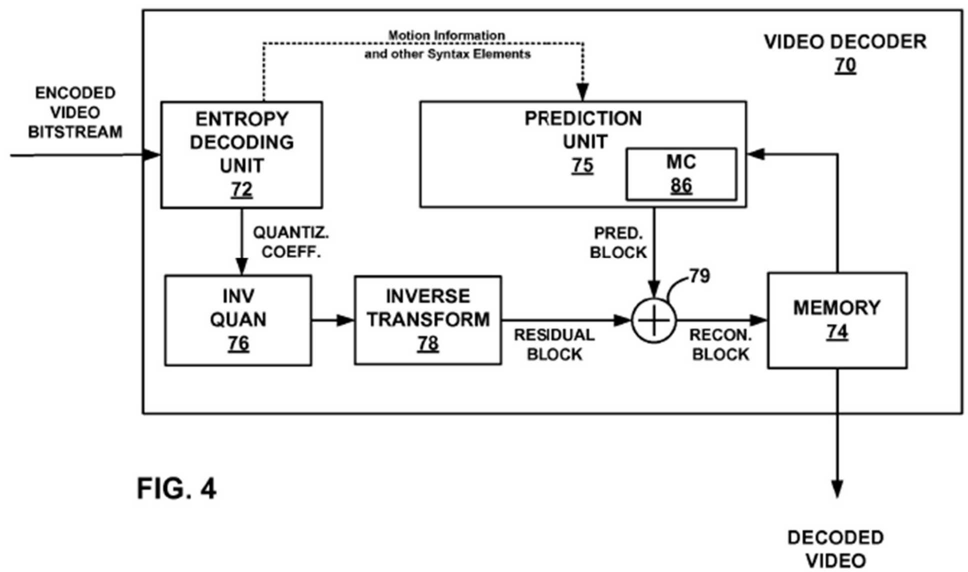
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, ¶0031:<br><br>The illustrated system 10 of FIG. 1 is merely exemplary. The non-zero rounding and prediction mode selection techniques of this disclosure may be performed by any coding device that supports bi-directional motion compensated prediction. Source device 12 and destination device 16 are merely examples of such coding devices in which source device 12 generates coded video data for transmission to destination device 16. In some cases, devices 12, 16 may operate in a substantially symmetrical manner such that, each of devices 12, 16 includes video encoding and decoding components. Hence, system 10 may support one-way or two-way video transmission between video devices 12, 16, e.g., for video streaming, video playback, video broadcasting, or video telephony.<br><br>Karczewicz I, ¶0035:<br><br>Video encoder 22 and video decoder 28 may operate according to a video compression standard, such as the ITU-T H.264 standard, alternatively described as MPEG-4, Part 10, Advanced Video Coding (AVC). The techniques of this disclosure, however, are not limited to any particular coding standard. Although not shown in FIG. 1, in some aspects, video encoder 22 and video decoder 28 may each be integrated with an audio encoder and decoder, and may include appropriate MUX-DEMUX units, or other hardware and software, to handle encoding of both audio and video in a common data stream or separate data streams. If |

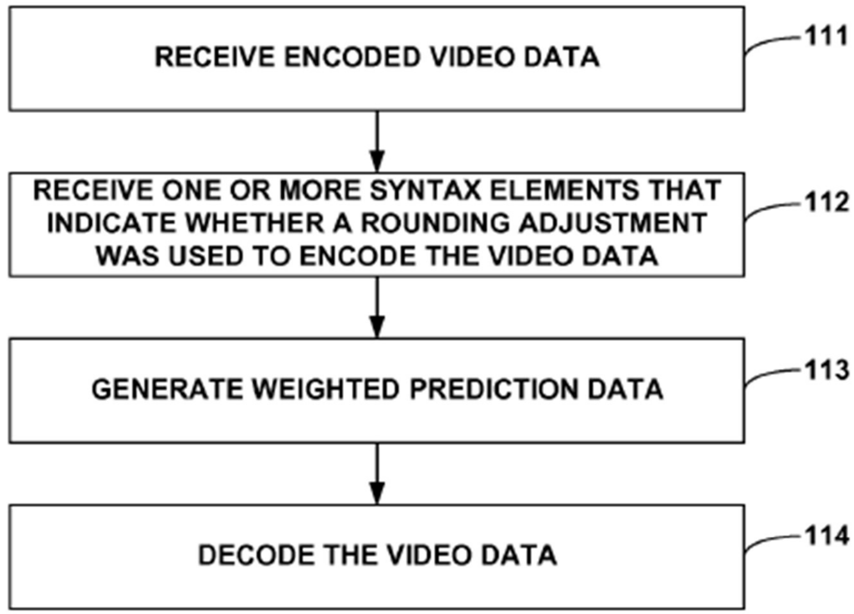
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | applicable, MUX-DEMUX units may conform to the ITU H.223 multiplexer protocol, or other protocols such as the user datagram protocol (UDP).<br><br>Karczewicz I, ¶0089:<br><br>Prediction unit 75 invokes motion compensation unit 86 for block based predictive decoding. In doing so, motion compensation unit 86 generates weighted prediction data based on the syntax (113). Accordingly, if the one or more syntax elements indicate that a rounding adjustment was used, then motion compensation unit 86 generates weighted prediction data that includes the rounding adjustment. However, if the one or more syntax elements indicate that a rounding adjustment was not used, then motion compensation unit 86 generates weighted prediction data that lacks the rounding adjustment. Video decoder 70 can then decode the video data using the weighted prediction data (114). In particular, video decoder 70 may invoke adder 79 to combine weighted prediction data (e.g., a prediction block) with residual video data (e.g., a residual block) in order to generate a reconstruction of the video data (e.g., a reconstructed video block).<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[c] using said second reference block to obtain a second prediction, said second prediction having the second precision; | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision.<br><br>*See, e.g.,* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
| --- | --- |
| | *See limitation 1[c].*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[d] obtaining a combined prediction based at least partly upon said first prediction and said second prediction; | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined prediction based at least partly upon the first prediction and the second prediction.<br><br>*See, e.g.,*<br><br>*See limitation 1[d].*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[e] decreasing a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br><br>*See, e.g.,* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *See limitation 1[e].*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 19[f] reconstructing the block of pixels based on the combined prediction. | Karczewicz I, as evidenced by the example citations below, discloses reconstructing the block of pixels based on the combined prediction.<br><br>*See, e.g.,*<br><br>Karczewicz I, ¶0086:<br><br>The quantized coefficients are sent from entropy decoding unit 72 to inverse quantization unit 76, which performs inverse quantization. Inverse transform unit 78 then inverse transforms the de-quantized coefficients back to the pixel domain to generate a residual block. Adder 79 combines the prediction data (e.g., a prediction block) generated by prediction unit 75 with the residual block from inverse transform unit 78 to create a reconstructed video block, which may be stored in memory 74 and/or output from video decoder 70 as decoded video output.<br><br>Karczewicz I, ¶0089:<br><br>Prediction unit 75 invokes motion compensation unit 86 for block based predictive decoding. In doing so, motion compensation unit 86 generates weighted prediction data based on the syntax (113). Accordingly, if the one or more syntax elements indicate that a rounding adjustment was used, then motion compensation unit 86 generates weighted prediction data |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | that includes the rounding adjustment. However, if the one or more syntax elements indicate that a rounding adjustment was not used, then motion compensation unit 86 generates weighted prediction data that lacks the rounding adjustment. Video decoder 70 can then decode the video data using the weighted prediction data (114). In particular, video decoder 70 may invoke adder 79 to combine weighted prediction data (e.g., a prediction block) with residual video data (e.g., a residual block) in order to generate a reconstruction of the video data (e.g., a reconstructed video block).<br><br>Karczewicz I, Fig. 4: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 4<br><br>Karczewicz I, Fig. 6: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 6 |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 20. The method according to claim 19, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel values of the first reference block.<br>*See, e.g.,*<br>*See claim 2.*<br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 21. The method according to claim 20, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter using values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br>*See, e.g.,*<br>*See claim 3.* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 22. The method according to claim 20, wherein in an instance in which said second motion vector points to an integer sample, said second prediction is obtained by shifting values of said second reference block to the left. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values of the second reference block to the left.<br><br>*See, e.g.,*<br><br>*See claim 4.*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 23. The method according to claim 19, wherein said decreasing said precision of said combined prediction by shifting bits of the combined prediction to the right, further comprises: inserting a | Karczewicz I, as evidenced by the example citations below, discloses that the decreasing the precision of the combined prediction by shifting bits of the combined prediction to the right, further comprises inserting a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.,* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| rounding offset to the combined prediction before said decreasing. | *See claim 5.*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 24. The method according to claim 19, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of said first prediction and values of said second prediction. | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of the first prediction and values of the second prediction.<br><br>*See, e.g.,*<br><br>*See claim 6.*<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 25[preamble] An apparatus for decoding a block of pixels, the apparatus comprising: | Karczewicz I, as evidenced by the example citations below, discloses an apparatus for decoding a block of pixels.<br><br>*See, e.g.,* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *See limitation 19[preamble].*<br><br>Karczewicz I, ¶0003:<br><br>Digital multimedia capabilities can be incorporated into a wide range of devices, including digital televisions, digital direct broadcast systems, wireless communication devices, wireless broadcast systems, personal digital assistants (PDAs), laptop or desktop computers, digital cameras, digital recording devices, video gaming devices, video game consoles, cellular or satellite radio telephones, digital media players, and the like. Digital multimedia devices may implement video coding techniques, such as MPEG-2, ITU-H.263, MPEG-4, or ITU-H.264/MPEG-4 Part 10, Advanced Video Coding (AVC), to transmit and receive or store and retrieve digital video data more efficiently. Video encoding techniques may perform video compression via spatial and temporal prediction to reduce or remove redundancy inherent in video sequences.<br><br>Karczewicz I, ¶0012:<br><br>The techniques described in this disclosure may be implemented in hardware, software, firmware, or any combination thereof If implemented in software, the software may be executed in one or more processors, such as a microprocessor, application specific integrated circuit (ASIC), field programmable gate array (FPGA), or digital signal processor (DSP). The software that executes the techniques may be initially stored in a computer-readable medium and loaded and executed in the processor.<br><br>Karczewicz I, ¶0038:<br><br>Video encoder 22 and video decoder 28 each may be implemented as one or more |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | microprocessors, digital signal processors (DSPs), application specific integrated circuits (ASICs), field programmable gate arrays (FPGAs), discrete logic, software, hardware, firmware or any combinations thereof. Each of video encoder 22 and video decoder 28 may be included in one or more encoders or decoders, either of which may be integrated as part of a combined encoder/decoder (CODEC) in a respective mobile device, subscriber device, broadcast device, server, or the like.<br><br>Karczewicz I, ¶0098:<br><br>The techniques of this disclosure may be implemented in a wide variety of devices or apparatuses, including a wireless handset, and integrated circuit (IC) or a set of ICs (i.e., a chip set). Any components, modules or units have been described provided to emphasize functional aspects and does not necessarily require realization by different hardware units. The techniques described herein may also be implemented in hardware, software, firmware, or any combination thereof. Any features described as modules, units or components may be implemented together in an integrated logic device or separately as discrete but interoperable logic devices. In some cases, various features may be implemented as an integrated circuit device, such as an integrated circuit chip or chipset.<br><br>Karczewicz I, ¶0099:<br><br>If implemented in software, the techniques may be realized at least in part by a computer-readable medium comprising instructions that, when executed in a processor, performs one or more of the methods described above. The computer-readable medium may comprise a computer-readable storage medium and may form part of a computer program product, which may include packaging materials. The computer-readable storage medium may comprise random access memory (RAM) such as synchronous dynamic random access memory (SDRAM), read-only memory (ROM), non-volatile random access memory (NVRAM), |

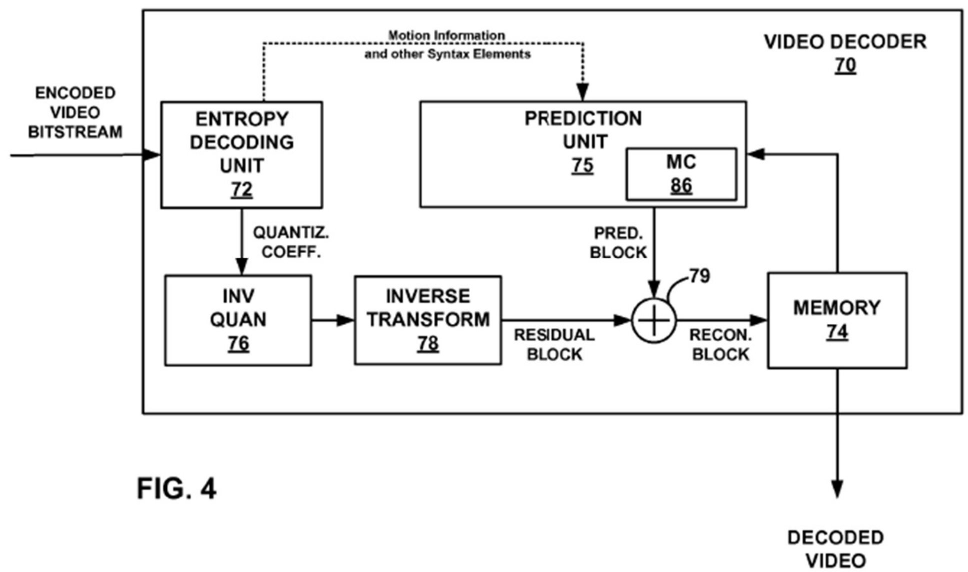
| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | electrically erasable programmable read-only memory (EEPROM), FLASH memory, magnetic or optical data storage media, and the like. The techniques additionally, or alternatively, may be realized at least in part by a computer-readable communication medium that carries or communicates code in the form of instructions or data structures and that can be accessed, read, and/or executed by a computer.<br><br>Karczewicz I, ¶0100:<br><br>The code or instructions may be executed by one or more processors, such as one or more digital signal processors (DSPs), general purpose microprocessors, an application specific integrated circuits (ASICs), field programmable logic arrays (FPGAs), or other equivalent integrated or discrete logic circuitry. Accordingly, the term "processor," as used herein may refer to any of the foregoing structure or any other structure suitable for implementation of the techniques described herein. In addition, in some aspects, the functionality described herein may be provided within dedicated software modules or hardware modules configured for encoding and decoding, or incorporated in a combined video codec. Also, the techniques could be fully implemented in one or more circuits or logic elements.<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 25[a] at least one processor and at least one memory including computer program code, the at least one memory | Karczewicz I, as evidenced by the example citations below, discloses that at least one processor and at least one memory including computer program code, the at least one memory and computer program code. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| and computer program code configured to, with the at least one processor, cause the apparatus to: | *See, e.g.*,<br><br>Karczewicz I, ¶0005:<br><br>For inter-coding, a video encoder performs motion estimation to track the movement of matching video blocks between two or more adjacent frames or other coded units, such as slices of frames. Inter-coded frames may include predictive frames ("P-frames"), which may include blocks predicted from a previous frame, and bidirectional predictive frames ("B-frames"), which may include blocks predicted from a previous frame and a subsequent frame of a video sequence. The terms P-frames and B-frames are somewhat historic in the sense that early coding techniques limited prediction in specific directions. Newer coding formats and standards may not limit the prediction direction of P-frames or B-frames. Thus, the term "bi-directional" now refers to prediction based on two or more lists of reference data regardless of the temporal relationship of such reference data relative to the data being coded.<br><br>Karczewicz I, ¶0006:<br><br>Consistent with newer video standards such as ITU H.264, for example, bi-directional prediction may be based on two different lists which do not necessarily need to have data that resides temporally before and after the current video block. In other words, B-video blocks may be predicted from two lists of data, which may correspond to data from two previous frames, two subsequent frames, or one previous frame and one subsequent frame. In contrast, P-video blocks are predicted based on one list, i.e., one data structure, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include only I-video blocks. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0022:<br><br>This disclosure describes video encoding and decoding techniques applicable to bi-directional prediction. In bi-directional prediction, a video block is predictively encoded and decoded based on two different lists of predictive reference data. In one aspect of this disclosure, rounding adjustments to bi-directional predictive data may be purposely eliminated to provide predictive data that lacks any rounding bias. In this case, rounded and unrounded predictive data may both be considered in a rate-distortion analysis to identify the best data for prediction of a given video block. One or more syntax elements may be encoded to indicate the selection, and a decoder may interpret the one or more syntax elements in order to determine whether rounding should be used in the decoding process.<br><br>Karczewicz I, ¶0042:<br><br>The techniques of this disclosure are specifically applicable to weighted bi-directional prediction. As mentioned above, bi-directional prediction is prediction of so-called "B-video blocks" based on two different lists of data. B-video blocks may be predicted from two lists of data from two previous frames, two lists of data from subsequent frames, or one list of data from a previous frame and one from a subsequent frame. In contrast, P-video blocks are predicted based on one list, which may correspond to one predictive frame, e.g., one previous frame or one subsequent frame. B-frames and P-frames may be more generally referred to as P-units and B-units. P-units and B-units may also be realized in smaller coded units, such as slices of frames or portions of frames. B-units may include B-video blocks, P-video blocks or I-video blocks. P-units may include P-video blocks or I-video blocks. I-units may include only I-video blocks. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | Karczewicz I, ¶0007:<br><br>For P- and B-video blocks, motion estimation generates motion vectors, which indicate the displacement of the video blocks relative to corresponding prediction video blocks in predictive reference frame(s) or other coded units. Motion compensation uses the motion vectors to generate prediction video blocks from the predictive reference frame(s) or other coded units. After motion compensation, a residual video block is formed by subtracting the prediction video block from the original video block to be coded. The video encoder usually applies transform, quantization and entropy coding processes to further reduce the bit rate associated with communication of the residual block. I-and P-units are commonly used to define reference blocks for the inter-coding of P- and B-units.<br><br>Karczewicz I, ¶0085:<br><br>In general, entropy decoding unit 72 receives an encoded bitsteam and entropy decodes the bitstream to generate quantized coefficients, motion information and other syntax. The motion information (e.g., motion vectors) and other syntax are forwarded to prediction unit 75 for use in generating the predictive data. Prediction unit 75 performs bidirectional prediction consistent with this disclosure, avoiding rounding adjustments in some cases, and possibly implementing default, implicit or explicit weighted prediction according to the received syntax elements. The syntax elements may identify the type of weighted prediction that to be used, coefficients and offset if explicit weighted prediction to be used, and whether rounding adjustments should be used in the decoding.<br><br>Karczewicz I, ¶0088:<br><br>FIG. 6 is a flow chart illustrating an exemplary process performed by a video decoder consistent with this disclosure. FIG. 6 will be described from the perspective of video decoder |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | 70 of FIG. 4. As shown in FIG. 6, video decoder receives encoded video data (111), and receives one or more syntax elements that indicate whether a rounding adjustment was used to encode the video data (1 12). In particular, entropy decoding unit 72 may receive an encoded bitstream that includes the video data and the one or more syntax elements. Following entropy decoding, entropy decoding unit 72 may output the video data as quantized transform coefficients, which are inverse quantized by unit 76 and inverse transformed by unit 78. Entropy decoding unit 72 may output syntax elements to prediction unit, which includes the one or more syntax elements that indicate whether a rounding adjustment was used to encode the video data, motion vectors and possibly other syntax.<br><br>Karczewicz I, Fig. 4: |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | <br>FIG. 4<br><br>Karczewicz I, ¶0053:<br><br>During the encoding process, video encoder 50 receives a video block to be coded, and motion estimation unit 32 and motion compensation unit 35 perform inter-predictive coding. Motion estimation unit 32 and motion compensation unit 35 may be highly integrated, but are illustrated separately for conceptual purposes. Motion estimation is typically considered the process of generating motion vectors, which estimate motion for video blocks. A motion vector, for example, may indicate the displacement of a predictive block within a predictive |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | frame (or other coded unit) relative to the current block being coded within the current frame (or other coded unit). Motion compensation is typically considered the process of fetching or generating the predictive block based on the motion vector determined by motion estimation. Again, motion estimation unit 32 and motion compensation unit 35 may be functionally integrated. For demonstrative purposes, the techniques described in this disclosure are described as being performed by motion compensation unit 35.<br><br>Karczewicz I, ¶0054:<br><br>Motion estimation unit 32 selects the appropriate motion vector for the video block to be coded by comparing the video block to video blocks of one or more predictive coded units (e.g., a previous and/or future frame in terms of time or temporally). Motion estimation unit 32 may, as an example, select a motion vector for a B-frame in a number of ways. In one way, motion estimation unit 32 may select a previous or future frame from a first set of frames (referred to as list 0) and determine a motion vector using only this previous or future frame from list 0. Alternatively, motion estimation unit 32 may select a previous or future frame from a second set of frames (referred to as list 1) and determine a motion vector using only this previous or future frame from list 1. In yet another way, motion estimation unit 32 may select a first frame from list 0 and a second frame from list 1 and select one or more motion vectors from the first frame of list 0 and the second frame of list 1. This third form of prediction may be referred to as bi-predictive motion estimation. Techniques of this disclosure may be implemented so as to efficiently select a motion-compensated bi-prediction mode. The selected motion vector for any given list may point to a predictive video block that is most similar to the video block being coded, e.g., as defined by a metric such as sum of absolute difference (SAD) or sum of squared difference (SSD) of pixel values of the predictive block relative to pixel values of the block being coded. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 25[b] determine, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision.<br><br>*See, e.g.*,<br><br>*See limitation 19[a]*. |
| 25[c] use said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br><br>*See, e.g.*,<br><br>*See limitation 19[b]*. |
| 25[d] use said second reference block to obtain a second prediction, said second prediction having the second | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| precision; | *See, e.g.*,<br><br>*See limitation 19[c]*. |
| 25[e] obtain a combined prediction based at least partly upon said first prediction and said second prediction; | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined prediction based at least partly upon the first prediction and the second prediction.<br><br>*See, e.g.*,<br><br>*See limitation 19[d]*. |
| 25[f] decrease a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br><br>*See, e.g.*,<br><br>*See limitation 19[e]*. |
| 25[g] reconstruct the block of pixels based on the combined prediction. | Karczewicz I, as evidenced by the example citations below, discloses reconstructing the block of pixels based on the combined prediction.<br><br>*See* limitation 19[f]. |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| 26. The apparatus according to claim 25, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel values of the first reference block.<br><br>*See, e.g.*,<br><br>*See claim 20.* |
| 27. The apparatus according to claim 26, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter using values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br><br>*See, e.g.*,<br><br>*See claim 21*. |
| 28. The apparatus according to claim 26, wherein in an instance in which said second motion vector points to an integer sample, said second prediction is obtained by shifting values of said second reference block to the left. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values of the second reference block to the left.<br><br>*See, e.g.*,<br><br>*See claim 22.* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| 29. The apparatus according to claim 25, wherein the at least one memory and computer code are configured to cause the apparatus to decrease said precision of said combined prediction by shifting bits of the combined prediction to the right, by: inserting a rounding offset to the combined prediction before said decreasing. | Karczewicz I, as evidenced by the example citations below, discloses that the at least one memory and computer code are configured to cause the apparatus to decrease the precision of the combined prediction by shifting bits of the combined prediction to the right, by inserting a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.*,<br><br>*See claim 23*,<br><br>*Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Karczewicz I, it would have been obvious in view of the disclosures in Karczewicz I, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit E-9* |
| 30. The apparatus according to claim 25, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of said first prediction and values of said second prediction. | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of the first prediction and values of the second prediction.<br><br>*See, e.g.*,<br><br>*See claim 24*. |
| 31[preamble] A computer program product for decoding a block of pixels, the | Karczewicz I, as evidenced by the example citations below, discloses a computer program product for decoding a block of pixels, the computer program product comprising at least one non-transitory computer readable storage medium having computer executable program code portions stored |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| computer program product comprising at least one non-transitory computer readable storage medium having computer executable program code portions stored therein, the computer executable program code portions comprising program code instructions configured to: | therein.<br><br>*See, e.g.*,<br><br>*See limitations 19[preamble], 25[preamble], 25[a].*<br><br>Karczewicz I, ¶0013:<br><br>> Accordingly, this disclosure also contemplates a computer-readable storage medium comprising instructions that when executed cause a processor to select between default weighted prediction and implicit weighted prediction, upon selecting default weighted prediction, select between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the default weighted prediction, upon selecting implicit weighted prediction, select between explicit weighted prediction based on calculated weights and explicit weighted prediction based on weights defined by the implicit weighted prediction, and encode the video data using weighed prediction data defined by the selections. |
| 31[a] determine, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision; | Karczewicz I, as evidenced by the example citations below, discloses determining, for a current block, a first reference block based on a first motion vector and a second reference block based on a second motion vector, wherein the pixels of the current block, the first reference block, and the second reference block have values with a first precision.<br><br>*See, e.g.*,<br><br>*See limitation 19[a].* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE:<br>KARCZEWICZ I |
|---|---|
| 31[b] use said first reference block to obtain a first prediction, said first prediction having a second precision, which is higher than said first precision; | Karczewicz I, as evidenced by the example citations below, discloses using the first reference block to obtain a first prediction, the first prediction having a second precision, which is higher than the first precision.<br><br>*See, e.g.*,<br><br>*See limitation 19[b].* |
| 31[c] use said second reference block to obtain a second prediction, said second prediction having the second precision; | Karczewicz I, as evidenced by the example citations below, discloses using the second reference block to obtain a second prediction, the second prediction having the second precision.<br><br>*See, e.g.*,<br><br>*See limitation 19[c].* |
| 31[d] obtain a combined prediction based at least partly upon said first prediction and said second prediction; | Karczewicz I, as evidenced by the example citations below, discloses obtaining a combined prediction based at least partly upon the first prediction and the second prediction.<br><br>*See, e.g.*,<br><br>*See limitation 19[d].* |
| 31[e] decrease a precision of said combined prediction by shifting bits of the combined prediction to the right; and | Karczewicz I, as evidenced by the example citations below, discloses decreasing a precision of the combined prediction by shifting bits of the combined prediction to the right.<br><br>*See, e.g.*, |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| | *See limitation 19[e].* |
| 31[f] reconstruct the block of pixels based on the combined prediction. | Karczewicz I, as evidenced by the example citations below, discloses reconstructing the block of pixels based on the combined prediction.<br><br>*See, e.g.,*<br><br>*See limitation 19[f].* |
| 32. The computer program product according to claim 31, wherein in an instance in which said first motion vector points to a subpixel, said first prediction is obtained by interpolation using pixel values of said first reference block. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the first motion vector points to a subpixel, the first prediction is obtained by interpolation using pixel values of the first reference block.<br><br>*See, e.g.,*<br><br>*See claim 20.* |
| 33. The computer program product according to claim 32, wherein said first prediction is obtained by interpolation using values of said first reference block by: right shifting a sum of a P-tap filter | Karczewicz I, as evidenced by the example citations below, discloses that the first prediction is obtained by interpolation using values of the first reference block by right shifting a sum of a P-tap filter using values of the first reference block.<br><br>*See, e.g.,* |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| using values of said first reference block. | *See claim 21.* |
| 34. The computer program product according to claim 32, wherein in an instance in which said second motion vector points to an integer sample, said second prediction is obtained by shifting values of said second reference block to the left. | Karczewicz I, as evidenced by the example citations below, discloses that in an instance in which the second motion vector points to an integer sample, the second prediction is obtained by shifting values of the second reference block to the left.<br><br>*See, e.g.,*<br><br>*See claim 22.* |
| 35. The computer program product according to claim 31, wherein the program code instructions configured to decrease said precision of said combined prediction by shifting bits of the combined prediction to the right, further comprise program code instructions configured to: insert a rounding offset to the combined prediction before said decreasing. | Karczewicz I, as evidenced by the example citations below, discloses that the program code instructions configured to decrease the precision of the combined prediction by shifting bits of the combined prediction to the right, further comprise program code instructions configured to insert a rounding offset to the combined prediction before the decreasing.<br><br>*See, e.g.,*<br><br>*See claims 23 and 29.* |
| 36. The computer program | Karczewicz I, as evidenced by the example citations below, discloses that the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the |

| ASSERTED CLAIMS | PRIOR ART REFERENCE: KARCZEWICZ I |
|---|---|
| product according to claim 31, wherein the first precision indicates a number of bits needed to represent the values of the pixels, and the second precision indicates the number of bits needed to represent values of said first prediction and values of said second prediction. | number of bits needed to represent values of the first prediction and values of the second prediction.<br><br>*See, e.g.,*<br><br>*See claim 24.* |