

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SNAP INC.,

Petitioner

v.

NOKIA TECHNOLOGIES OY,

Patent Owner.

IPR2025-01114

U.S. Patent No. 8,175,148

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 8,175,148**

I.	INTRODUCTION	1
II.	GROUND FOR STANDING	1
III.	IDENTIFICATION OF CHALLENGE.....	1
A.	Prior Art	1
B.	Grounds for Challenge.....	2
IV.	'148 PATENT OVERVIEW	2
V.	LEVEL OF ORDINARY SKILL.....	5
VI.	CLAIM CONSTRUCTION.....	6
A.	MPF.....	6
1.	“Means for Encoding” (claim 12)/“Module for Encoding” (claim 23)...	7
2.	“Means for Transforming” (claim 12) / “Module for Transforming” (claim 23).....	8
3.	“Means for Selecting” (claim 12) / “Module for Selecting” (claim 23)..	8
4.	“Means for Providing” (claim 12)	9
VII.	DETAILED EXPLANATION OF GROUNDS.....	9
A.	Ground 1: Obviousness over MPEG-1	9
1.	MPEG-1	9
2.	Claim 1 and Dependents	13
3.	Claim 12 and Dependents	28
4.	Claim 23	43
B.	Ground 2: Obviousness over H.263	45
1.	H.263.....	45
2.	Claim 1 and Dependents	47
3.	Claim 12 and Dependents	71
VIII.	CONCLUSION.....	88
IX.	MANDATORY NOTICES.....	88
A.	Real Party in Interest.....	88
B.	Related Matters	89

C.	Notice of Counsel and Service Information	90
----	---	----

EXHIBIT LIST

Exhibit	Reference
1001	U.S. Patent No. 8,175,148 (“’148 Patent” or “’148Pat.”)
1002	File History of the ’148 Patent (“’148FH”)
1003	Declaration of Dr. Eli Saber (“Saber”)
1004	Dr. Eli Saber CV
1005	ISO/IEC 11172-2 (1st ed.) Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s - Part 2: Video (August 1, 1993) (“MPEG-1”)
1006	ITU-T Recommendation H.263 (February 1998) (“H.263”)
1007	Intentionally Left Blank
1008	ITU-T Recommendation H.324 (February 1998) (“H.324”)
1009	ITU-T Recommendation H.264 (May 2003) (“H.264”)
1010	Declaration of Sylvia Hall-Ellis (“Hall-Ellis”)
1011	Bhaskaran, Vasudev, Konstantinides, Konstantinos, Image and Video Compression Standards, Algorithms and Architectures, (Kluwer Academic Publishers, 1995) (“Bhaskaran”)
1012	ITU-T Recommendation H.261 (March 1993) (“H.261”)

TABLE OF ABBREVIATIONS

Abbreviation	Definition
'148 Patent/'148Pat.	U.S. Patent 8,175,148 (EX-1001)
Challenged Claims	Claims 1–23
DCT	Discrete Cosine Transform
IPR	<i>inter partes</i> review
POSITA	Person of Ordinary Skill in the Art
Snap/Petitioner	Snap Inc.

LISTING OF CHALLENGED CLAIMS

Reference	Claim Limitation
Claim 1	
1[pre]	A method of encoding a digital video sequence for use in a video coding application to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels, said method comprising:
1[a]	encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels for producing corresponding blocks of prediction error values;
1[b]	applying a transform coding technique to said blocks of prediction error values to produce sets of transform coefficient values representative of said blocks of prediction error values;
1[c]	defining a default level of quantization for use in encoding of the digital video sequence to quantize the sets of transform coefficient values; and
1[d]	providing an indication of the default level of quantization to a decoding process.
Claim 2	
2	An encoding method according to claim 1, wherein said indication of the default level of quantization is provided in the encoded bitstream representative of the digital video sequence.
Claim 3	
3	An encoding method according to claim 2, wherein the encoded bitstream is transmitted from a video encoding device to a corresponding video decoding device.

Claim 4	
4	An encoding method according to claim 1, wherein the default level of quantization is updated during the encoding of the digital video sequence, said method further comprising providing an indication of the updated default level of quantization to a decoding process.
Claim 5	
5	An encoding method according to claim 1, wherein said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from the default level of quantization.
Claim 6	
6	An encoding method according to claim 5, further comprising providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization.
Claim 7	
7	An encoding method according to claim 5, wherein said level of quantization is adjusted from one frame of the digital video sequence to another such that an actual level of quantization applied to the sets of transform coefficients for a particular frame of the digital video sequence is different from the default level of quantization.
Claim 8	
8	An encoding method according to claim 7, wherein said actual level of quantization is representable as a difference with respect to the default level of quantization.
Claim 9	

9	An encoding method according to claim 7, further comprising providing to a decoding process an indication of a difference between said actual level of quantization and the default level of quantization.
Claim 10	
10	An encoding method according to claim 5, wherein the plurality of blocks of pixels, into which a frame of said digital video sequence is divided, are grouped into one or more segments and wherein the level of quantization applied to said sets of transform coefficient values is adjusted from one segment of a frame to another such that an actual level of quantization applied to the sets of transform coefficients for a particular segment of a frame is different from the default level of quantization.
Claim 11	
11	An encoding method according to claim 10, further comprising providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization.
Claim 12	
12[pre]	A video encoder for encoding a digital video sequence to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels, said video encoder comprising:
12[a]	means for encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels to provide corresponding blocks of prediction error values;
12[b]	means for transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values;

12[c]	means for selecting a default level of quantization for quantizing said sets of transform coefficient values; and
12[d]	means for providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence.
Claim 13	
13	A video encoder according to claim 12, wherein said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from the default level of quantization.
Claim 14	
14	A video encoder according to claim 13, arranged to provide an indication of a difference between said level of quantization and the default level of quantization in the encoded bit-stream.
Claim 15	
15	A video encoder according to claim 12, wherein the default level of quantization can be updated and the video encoder is arranged to provide an indication of the updated default level of quantization in the encoded bit-stream.
Claim 16	
16	A video encoder according to claim 15, wherein said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from the updated default level of quantization.
Claim 17	
17	A video encoder according to claim 12, provided in a multimedia terminal.
Claim 18	

18	A video encoder according to claim 12, provided in a radio telecommunications device.
Claim 19	
19	A video encoder according to claim 13, wherein said level of quantization applied to said sets of transform coefficient values can be adjusted from one frame of the digital video sequence to another so as to apply an actual level of quantization to the sets of transform coefficients for a particular frame of the digital video sequence and wherein said actual level of quantization is different from the default level of quantization.
Claim 20	
20	A video encoder according to claim 19, arranged to provide an indication of said difference with respect to the default level of quantization in the encoded bit-stream representative of the digital video sequence.
Claim 21	
21[a]	A video encoder according to claim 13, arranged to group the plurality of blocks into which a frame of said digital video sequence is divided into one or more segments and
21[b]	to adjust the level of quantization applied to said sets of transform coefficient values from one segment of a frame to another
21[c]	so as to apply an actual level of quantization to the sets of transform coefficients for a particular segment of a frame, wherein said actual level of quantization is different from the default level of quantization,
21[d]	said video encoder further arranged to represent the actual level of quantization for use in the particular segment as a difference with respect to the default level of quantization.
Claim 22	
22	A video encoder according to claim 21, further arranged to provide an indication of said difference with respect to the default level of

	quantization in the encoded bit-stream representative of the digital video sequence.
Claim 23	
23[pre]	A video encoder for encoding a digital video sequence to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels, said video encoder comprising:
23[a]	a module for encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels to provide corresponding blocks of prediction error values;
23[b]	a module for transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values; and
23[c]	a module for selecting a default level of quantization for quantizing said sets of transform coefficient values, wherein the default level is specific to the video coding applications.

I. INTRODUCTION

Snap requests IPR of the Challenged Claims of the '148 Patent.

II. GROUNDS FOR STANDING

Petitioner certifies the '148 Patent is available for IPR, and Petitioner is not barred or estopped from requesting IPR on the grounds herein. An IPR was previously filed and instituted for the '148 patent but subsequently terminated due to a settlement between the parties. *See* IPR2024-01176, paper no. 13. This petition is substantially similar to the previously instituted petition. *See* IPR2024-01176, paper no. 3. Snap had no involvement, and was not a real party in interest, in the previous IPR.

III. IDENTIFICATION OF CHALLENGE

A. Prior Art

Petitioner respectfully requests cancellation of the Challenged Claims based on:

Prior Art
MPEG-1; publicly available no later than October 21, 1993 ¹ ; prior art under at least 35 U.S.C. §102(b) ² (EX-1005).

¹ EX-1010.

² All references to 35 U.S.C. §§102, 103 are to the pre-AIA statutory framework.

Prior Art	
H.263; ratified February 6, 1998 and publicly available no later than May 11, 2000 ³ ; prior art under at least 35 U.S.C. §102(b) (EX-1006).	

B. Grounds for Challenge

Ground		Claims	Prior Art
1	§103	1-23	MPEG-1
2		1-22	H.263

IV. '148 PATENT OVERVIEW

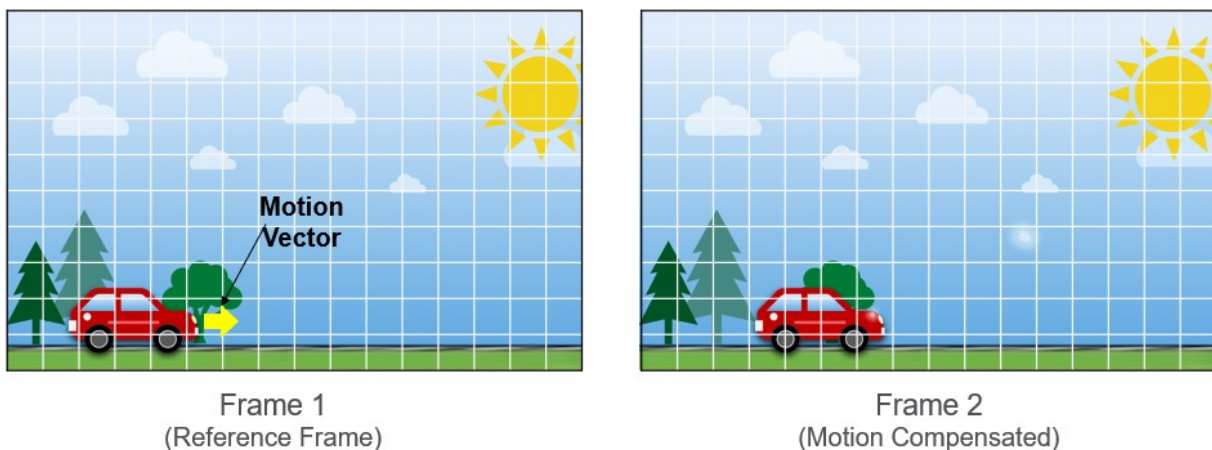
The '148 Patent issued on May 8, 2012, from U.S. Patent Application 11/881,367, filed July 26, 2007. The '148 Patent claims priority to U.S. Provisional Application 60/374,667, filed April 23, 2002.

The patent is directed to providing a quantization parameter indication, for decoding purposes, in an encoded bit-stream for a video sequence. '148Pat., Abstract. Large amounts of data are required to transmit uncompressed video, so compression techniques have long been used to reduce redundancies. '148Pat., 2:2534; 2:45-51. One technique is “‘block-based transform coding’, in which a mathematical transformation, such as a two-dimensional [DCT], is applied to blocks of image pixels. This transforms the image data from a representation comprising

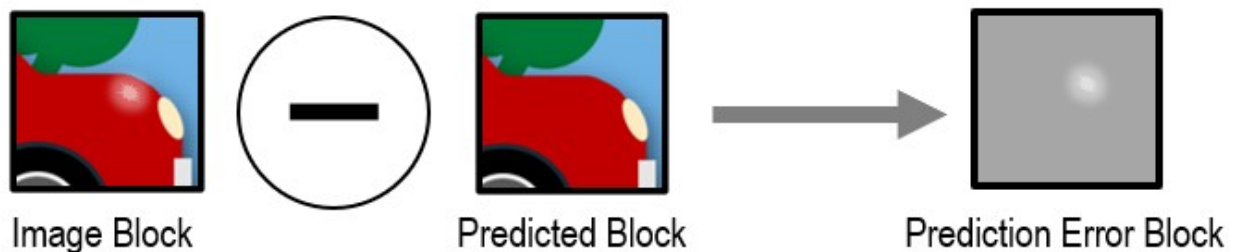
³ EX-1010.

pixel values to a form comprising a set of coefficient values representative of spatial frequency components,” reducing spatial redundancy in a frame. ’148Pat., 2:60-3:7. “Frames of a video sequence which are compressed using block-based transform coding, without reference to any other frame within the sequence, are referred to as INTRA-coded.” ’148Pat., 3:8-11.

Another technique is “motion-compensated prediction,” which reduces a video’s temporal redundancy by predicting a frame using “motion vectors” that characterize motion of objects in other, previously coded, “reference” frames. ’148Pat., 3:12-22. Motion-compensated prediction may be performed on individual portions of a frame, known as “macroblocks.” ’148Pat., 1:51-56, 3:23-25. “Frames of a video sequence, compressed using motion-compensated prediction, are generally referred to as INTER-coded.” ’148Pat., 3:27-29. This is illustrated below, a macroblock in Frame 2 is predicted using a motion vector characterizing the motion of a macroblock in Frame 1. Saber, ¶¶44-45.



“Motion-compensated prediction alone rarely provides a sufficiently precise representation of the image content of a video frame and therefore it is typically necessary to provide a so-called ‘prediction error’ (PE) frame with each INTERcoded frame. The [PE] frame represents the difference between a decoded version of the INTER-coded frame and the image content of the frame to be coded.” ’148Pat., 3:29-36; Saber, ¶¶46-47.



Block-based transform coding (e.g., DCT) can subsequently be applied—producing coefficients for each PE block—to reduce spatial redundancy. ’148Pat., 3:40-44, 6:10-15. Further, DCT coefficients are quantized “by dividing each coefficient value by the quantization parameter QP and rounding the result to the nearest integer,” yielding quantized DCT coefficient values with reduced numerical precision compared with the original (e.g., unquantized) DCT coefficient values. ’148Pat., 4:21-42, 4:47-50, 6:16-20. Increasing QP provides a more compressed video at the expense of quality, while decreasing QP provides higher quality but less compressed video. ’148Pat., 4:52-62, 6:22-27.

A decoder must know the QP value the encoder used for quantization, and that value is typically transmitted once per “slice” (which contains part of a frame and is formed from macroblocks in consecutive coding order). ’148Pat., 2:5-11, 10:33-40. Prior-art solutions like H.26L transmitted the QP value for each picture, and optionally for each slice forming the picture. ’148Pat., 10:44-60. In H.26L, the QP value could also be adjusted per-macroblock using a quantizer change parameter. ’148Pat., 11:8-14.

The ’148 Patent purportedly improves upon prior art by introducing a sequence-level quantization parameter (SQP) transmitted by an encoder in a bitstream. ’148Pat., 11:25-27, 18:9-16, 19:60-67. The SQP provides “a default or reference level of quantization,” used throughout encoding of the video. ’148Pat., 19:4-11. Further, the actual QP can be adjusted. ’148Pat., 12:29-34. For example, the encoder can determine the QP for quantizing the DCT coefficients of a slice, as well as the difference (Δ QP) between the SQP and QP. ’148Pat., 20:28-40. The Δ QP is transmitted in the encoded header. ’148Pat., 20:40-46. A receiving decoder combines the Δ QP with a previously-received SQP to determine the quantization level applied to the slice’s DCT coefficients. ’148Pat., 22:6-11, Fig. 7.

V. LEVEL OF ORDINARY SKILL

A POSITA at the time of the ’148 Patent would have had: (1) bachelor’s degree in electrical/computer engineering, Computer Science, or comparable field

of study such as Physics, and (2) approximately 2-3 years practical experience with video and/or image processing or coding. More education could substitute for experience, and vice versa. Saber, ¶¶23-24.

VI. CLAIM CONSTRUCTION

Claims are construed “in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.” 37 C.F.R. §42.100(b). Petitioner is aware that the PTAB preliminarily determined claim 1 requires a sequence-level parameter. IPR2024-01176, paper no. 9.⁴ As described below, Petitioner proposes constructions for means-plus-function (MPF) limitations in certain Challenged Claims.

A. MPF⁵

When a claim uses “means” or “module” and includes functional language, it is presumably MPF. *See* MPEP §2181. *TriMed, Inc. v. Stryker Corp.*, 514 F.3d 1256, 1259-60 (Fed. Cir. 2008). Claim 12 recites “*means for encoding...*,” “*means*

⁴ The Board only found this distinction relevant to H.263, not MPEG-1. *Id.*

⁵ Petitioner reserves the right to address indefiniteness, enablement, and/or written description issues in other proceedings.

for transforming...,” “means for selecting...,” and “means for providing...” Claim 23 recites “a module for encoding...,” “a module for transforming...,” and “a module for selecting...”

1. “Means for Encoding” (claim 12)/“Module for Encoding” (claim 23)

The function is “applying motion compensated prediction to blocks of pixels,” and “provid[ing] corresponding blocks of prediction error values.” ’148Pat., 26:51-54, 28:21-24; Saber, ¶¶64-66. The corresponding structure is a motion-compensated predictor and combiner. Saber, ¶67. The ’148 Patent’s motion-compensated predictor 150 (also called “motion compensated prediction block”) uses “the approximate representation of the motion vector generated by motion field coding block” and “generates an array of predicted pixel values for each block of the macroblock.” ’148Pat., 7:66-8:4. Further, “[e]ach block of predicted pixel values is passed to combiner 116 where the predicted pixel values are subtracted from the actual (input) pixel values in the corresponding block of the current macroblock, thereby forming a set of prediction error blocks for the macroblock.” ’148Pat., 8:5-9. The combiner provides blocks of prediction error values. Saber, ¶¶68-71.

2. “Means for Transforming” (claim 12) / “Module for Transforming” (claim 23)

The function is “providing sets of transform coefficient values representative of said blocks of prediction error values.” ’148Pat., 26:55-57, 28:25-28; Saber, ¶¶72-74. The corresponding structure is a transform block. Saber, ¶75. The ’148 Patent’s transform block 104 (also called “transform unit,” “DCT transformation block,” and “DCT transformer”) applies DCT to blocks of prediction error values to produce transform coefficients. ’148Pat., 18:66-19:3. Thus, transform block 104 provides transform coefficient values representative of blocks of prediction error values. Saber, ¶¶76-77.

3. “Means for Selecting” (claim 12) / “Module for Selecting” (claim 23)

The function of “means for selecting” is “selecting a default level of quantization for quantizing said sets of transform coefficient values.” ’148Pat., 26:58-59; Saber, ¶¶78-80. The function of “module for selecting” adds “wherein the default level is specific to the video coding applications” to the “means for selecting” function. ’148Pat., 28:29-32; Saber, ¶¶78-80.

The corresponding structure is a control manager. Saber, ¶81. The ’148 Patent’s control manager 660 selects the quantization parameter used to quantize the transform coefficients for each prediction error block. ’148Pat., 6:16-22; 19:60-

65. Thus, control manager 660 selects a default level of quantization for quantizing sets of transform coefficient values. Saber, ¶¶82-83.

4. “Means for Providing” (claim 12)

The function is “providing an indication of said default level of quantization in the encoded bit-stream.” ’148Pat., 26:60-62; Saber, ¶¶84-85. The corresponding structure is a video multiplex coder. Saber, ¶86. The ’148 Patent’s video multiplex coder 670 encodes control information (e.g., quantization information) in a bitstream. ’148Pat., 6:38-47, 19:60-65. Thus, video multiplex coder 670 provides an indication of an encoded bit-stream’s default level of quantization. Saber, ¶¶87-88.

VII. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Obviousness over MPEG-1

1. MPEG-1

MPEG-1 (EX-1005) is a video compression standard first ratified on August 1, 1993, and publicly available by October 1993. Hall-Ellis, 54-60. MPEG-1 describes syntax for a coded-video bitstream including six layers: (1) sequence; (2) group-of-pictures; (3) picture; (4) slice; (5) macroblock; and (6) block. MPEG-1, 16; Saber, ¶89.

MPEG-1 describes picture types, including intra-coded pictures (I-pictures) coded without reference to other pictures, and predictive-coded pictures (P-

pictures) coded using motion-compensated prediction from a past I-picture or P-picture. MPEG-1, 13; Saber, ¶90. The coding uses motion compensation, or “forward prediction,” to reduce temporal redundancy. MPEG-1, 13, 96; Saber, ¶90.

Motion compensation uses vectors to represent displacement between picture elements (“pels”)⁶ in a previously-transmitted picture and a predicted block. MPEG-1, 95. “The predicted block is usually a good estimate of the current block, and it is usually more efficient to transmit the motion vector plus the difference between the predicted block and the current block than to transmit a description of the current block by itself.” MPEG-1, 95. The difference between a current block and a predicted block is “prediction error.” MPEG-1, 13; Saber, ¶91. MPEG-1 compresses prediction error by (1) applying DCT to each 8-pel-by-8-line block of prediction error and (2) quantizing the result. MPEG-1, 13-14; Saber, ¶92. In step (1), DCT transforms prediction error into the DCT domain, producing DCT coefficients. MPEG-1, 14-15; Saber, ¶92. In step (2), quantization (representing a range of values using one value in the range) of the DCT coefficients produces quantized DCT coefficients that can be encoded using fewer bits. MPEG-1, 14, 69; Saber, ¶¶92-95.

⁶ “Pel” is interchangeable with “pixel.” Saber, ¶91.

MPEG-1 quantization uses a non-intra quantization matrix (“non_intra_quantizer_matrix”) included in a video’s sequence header.⁷ MPEG-1, 36. It is “a list of sixty-four 8-bit unsigned integers [that are] in effect until the next occurrence of a sequence header,”⁸ and each unsigned integer of the matrix is used to quantize a corresponding DCT coefficient of the block of DCT coefficients (e.g., produced from a block of the prediction error). MPEG-1, 43, 53; Saber, ¶¶96-98.

MPEG-1 quantization also uses “quantizer_scale” included in the video’s slice and/or macroblock layer syntax. MPEG-1, 39, 110-11. It is an “integer in the range [1–31] [that] scale[s] the reconstruction level of the retrieved DCT coefficient levels.” MPEG-1, 46. This means quantizer_scale multiplies the outcome of inverse quantization (“reconstruction level”) applied to a quantized coefficient (“retrieved DCT coefficient level”). Saber, ¶¶99-101. In MPEG-1 example code, “quantizer_scale” multiplies individual elements of the

⁷ MPEG-1 uses “quantizer matrix” and “quantization matrix” interchangeably and uses “non-intra quantization matrix” and “non_intra_quantizer_matrix” to refer to the same matrix. Saber, ¶96.

⁸ The sixty-four integers represent an 8x8 matrix, corresponding to an 8-pelby-8-line block. Saber, ¶98.

“non_intra_quant” matrix to reconstruct DCT coefficients at the decoder. MPEG-1, 53; Saber, ¶102.

```

for ( m=0; m<8; m++ ) {
    for ( n=0; n<8; n++ ) {
        i = scan[m][n] ;
        dct_recon[m][n] = ( ( (2 * dct_zz[i]) + Sign(dct_zz[i]) ) *
                             quantizer_scale * non_intra_quant[m][n] ) / 16 ;
        if ( ( dct_recon[m][n] & 1 ) == 0 )
            dct_recon[m][n] = dct_recon[m][n] - Sign(dct_recon[m][n]) ;
        if (dct_recon[m][n] > 2047) dct_recon[m][n] = 2047 ;
        if (dct_recon[m][n] < -2048) dct_recon[m][n] = -2048 ;
        if ( dct_zz[i] == 0 )
            dct_recon[m][n] = 0 ;
    }
}

```

Similarly, during encoding, each DCT coefficient “coefforig” is divided by the product (see the denominator of the above listed formula clearly showing “quantscale*niqmatrix”) of a corresponding element in “niqmatrix” and the “quantscale.” MPEG-1, 115; Saber, ¶103.

```

int coefforig;    /* original coefficient */
int coeffqant;    /* quantized coefficient */
int coeffrec;     /* reconstructed coefficient */
int niqmatrix;    /* non-intra quantization matrix */
int quantscale;   /* quantizer scale */

coeffqant = (8 * coefforig) / (quantscale * niqmatrix);

```

2. Claim 1 and Dependents

a. Claim 1

i. 1[pre]

If limiting, MPEG-1 discloses or suggests 1[pre]. Saber, ¶104. MPEG-1 includes a block diagram of an encoder and describes functions performed to produce “a valid coded bitstream” “for compressing video sequences” (*“a method of encoding a digital video sequence for use in a video coding application to produce an encoded video bit-stream representative of the digital video sequence”*). MPEG-1, 12, 14, 22; Saber, ¶105. MPEG-1 specifies “the syntax and semantics of the bitstream and the signal processing” used in encoding. MPEG1, 14. MPEG-1’s Annex D explains coding methods related to the decoding clauses of MPEG-1. MPEG-1, 68.

MPEG-1 uses the term “picture” “in preference to the terms field or frame.” MPEG-1, 25. A picture is therefore “*a frame.*” Saber, ¶106. An MPEG-1 “sequence” includes a series of pictures (*“the digital video sequence comprising a number of frames”*). MPEG-1, 27; Saber, ¶106. MPEG-1’s “group-of-pictures” or “GOP” is a series of coded pictures (*“each frame of said sequence comprising an array of pixels”*). MPEG-1, 32; Saber, ¶106. Each picture (consists of one or more slices (which do not overlap and have no gaps therebetween and each include at least one macroblock). MPEG-1, 34, 71. Each macroblock has 4 luminance blocks

(representing brightness) and 2 chrominance blocks (representing color difference) (“*divided into a plurality of blocks*”). MPEG-1, 34; Saber, ¶107. Further, each block (whether luminance or chrominance) represents an 8-pel-by-8-line section (“*each block comprising a certain number of said pixels*”). MPEG-1, 34; Saber, ¶107. MPEG-1 pictures thus comprise pixels that are divided into luminance and chrominance “*blocks*.”

ii. 1[a]

MPEG-1 discloses or suggests 1[a]. Saber, ¶108.

MPEG-1 discloses “*encoding*” using temporal processing of P-pictures “using motion compensated prediction from a past intra or predictive coded picture” (“*encoding a frame of the digital video sequence by applying motion compensated prediction*”). MPEG-1, 13; Saber, ¶¶109-10. P-pictures use forward motion compensation from the last transmitted I- or P-picture. MPEG-1, 96. Motion compensation “predicts the values of a block pels in a picture by relocating a block of neighboring pel values from a known [past] picture.” MPEG-1, 70. The prediction uses motion vectors providing offsets into the reference I- or P-picture containing the decoded pel values. MPEG-1, 24; Saber, ¶¶111-12. In P-pictures, one motion vector is used for each non-intra macroblock. MPEG-1, 14.

As explained in §VII.A.2.a(i), a MPEG-1 macroblock includes four blocks of luminance component and two blocks of chrominance component, and each

block is an 8-pel-by-8-line section (“a *block comprising a certain number of said pixels*” or a “*block of pixels*”). MPEG-1, 15, 34; Saber, ¶113. The one motion vector used for each macroblock—including the chrominance and luminance blocks—is applied to “*blocks of pixels*.” Saber, ¶113; MPEG-1, 13-14. Thus, by applying a motion vector for each non-intra macroblock, MPEG-1 “*applies motion compensated prediction to blocks of pixels*.” Saber, ¶113. The motion-compensated prediction uses those motion vectors to form prediction error signals for the respective blocks, “*producing corresponding blocks of prediction error values*.” MPEG-1, 24; Saber, ¶113.

iii. 1[b]

MPEG-1 discloses or suggests 1[b]. Saber, ¶114.

MPEG-1 uses DCT (“*a transform coding technique*”) to transform prediction error signals (“*said blocks of prediction error values*”). MPEG-1, 13; Saber, ¶115. Specifically, the same DCT transforms each 8-pel-by-8-line block of a predictive coded macroblock, where each block represents prediction error. MPEG-1, 13-15; Saber, ¶¶116-17. In MPEG-1, the difference between a motion-compensated macroblock and the current macroblock is transformed with a two-dimensional DCT to produce an array of transform coefficients (“*produc[ing] sets of transform coefficient values representative of said blocks of prediction error values*”). MPEG-1, 110; Saber, ¶117.

iv. 1[c]

MPEG-1 discloses or suggests 1[c]. Saber, ¶118.

In MPEG-1, each 8-pel-by-8-line “block of DCT coefficients” (“*the sets of transform coefficient values*”) is quantized “and scanned in a zig-zag order to convert it into a one-dimensional string of quantized DCT coefficients.” MPEG-1, 15; Saber, ¶¶119-20. For encoding sequences, a quantizer matrix quantizes DCT coefficients (“*to quantize the sets of transform coefficient values*”). MPEG-1, 83; Saber, ¶120.

MPEG-1’s one-bit flag (“load_non_intra_quantizer_matrix”) in the sequence header indicates the non_intra_quantizer_matrix used to quantize the DCT coefficients. MPEG-1, 36, 42; Saber, ¶¶120-22. When the flag is set to “0,” the non_intra_quantizer_matrix is “reset” to “default values”—an 8-pel-by-8-line matrix with all its values equal to 16. MPEG-1, 42, 83-84. This “default” non_intra_quantizer_matrix (of all 16s) is used to quantize the frames in a sequence “until the next occurrence of the sequence header.” MPEG-1, 42; Saber, ¶123.

When “load_non_intra_quantizer_matrix” is set to “1,” the flag is followed in the sequence header by “a list of sixty-four 8-bit unsigned integers” used in place of the “default” matrix. MPEG-1, 36, 42-43, 83. These values remain in effect until the next sequence header. MPEG-1, 43; Saber, ¶124.

Therefore, whether the non_intra_quantizer_matrix has been reset to its default value of all 16s (i.e., “load_non_intra_quantizer_matrix” is zero) or to the values provided in the sequence header (i.e., “load_non_intra_quantizer_matrix” is set to “1”), the non_intra_quantizer_matrix provides “*a default level of quantization.*” MPEG-1, 42, 43, 83; Saber, ¶125.

A subsequent sequence header can update MPEG-1’s non_intra_quantizer_matrix during encoding. This is consistent with the ’148 Patent, where the default level of quantization “used throughout encoding of the digital video sequence,” “can be updated during encoding of the digital video sequence.” ’148Pat., 12:19-24; Saber, ¶¶126-27.

Given the ’148 Patent’s use of a default level of quantization until it is updated (’148Pat., 12:19-24), a POSITA would have understood “defining” a default level of quantization in an encoder encompasses selecting or determining the level of quantization to be used throughout the sequence (i.e. for all pictures in the sequence) until the next update. Saber, ¶128. A POSITA would have further understood when an MPEG-1 encoder determines the value of “load_non_intra_quantizer_matrix,” discussed above, the encoder is selecting or determining a level of quantization. Saber, ¶128. Similarly, when an MPEG-1 encoder sets “load_non_intra_quantizer_matrix” to “1,” the values that follow in the sequence header (for the non_intra_quantizer_matrix) are levels of quantization

selected or determined by the encoder. Saber, ¶128. In either case, MPEG-1 “*define[s] a default level of quantization.*” Saber, ¶128.

v. 1[d]

MPEG-1 discloses or suggests 1[d]. Saber, ¶129.

MPEG-1’s sequence header is a block of data in the coded bitstream that includes elements for decoding a video sequence (“*providing...to a decoding process*”). MPEG-1, 26, 32. The sequence header provides information to the decoder to decode and play a video. Saber, ¶130. Parameters defining the quantizer matrices may be redefined each time a sequence header occurs in the bitstream. MPEG-1, 32.

As explained in §VII.A.2.a(iv), MPEG-1 discloses a flag (“load_non_intra_quantizer_matrix”) in the sequence header indicating how values for the non_intra_quantizer_matrix are to be specified. MPEG-1, 36, 42. The flag is “*an indication of the default level of quantization.*” Saber, ¶131. If PO argues a flag does not satisfy this limitation, MPEG-1 nevertheless discloses, when the “load_non_intra_quantizer_matrix” flag is set to “1,” it is followed in the sequence header by “a list of sixty-four 8-bit unsigned integers” used in place of the “default” values of all 16s. MPEG-1, 36, 42-43, 83. The sequence header thus includes in the bitstream the values of the non_intra_quantizer_matrix, which teaches “*an indication of the default level of quantization.*” Saber, ¶131. Further, a

decoder will decode the sequence header, including sequence parameters. MPEG-1, 120.

b. Claim 2

MPEG-1 discloses or suggests claim 2's limitation. Saber, ¶132.

As described in §VII.A.2.a(v), MPEG-1 discloses an indication of the non_intra_quantizer_matrix (“*the default level of quantization*”) provided in a sequence header. MPEG-1, 36, 42, 43; Saber, ¶133. The sequence header begins a coded video sequence (“*provided in the encoded bit-stream representative of the digital video sequence*”). MPEG-1, 32; Saber, ¶133.

c. Claim 3

MPEG-1 discloses or suggests claim 3's limitation. Saber, ¶134.

MPEG-1 describes an encoded bitstream's syntax and semantics and the signal processing in the decoder. MPEG-1, 26, 32; Saber, ¶135. Accordingly, MPEG-1 describes a video encoder that encodes the bitstream in accordance with the syntax and semantics of the standard and a video decoder that receives the coded video bitstream transmitted by the encoder (“*the encoded bit-stream is transmitted from...encoding device to a corresponding...decoding device*”). MPEG-1, 14, 16; Saber, ¶135. A POSITA would therefore have understood MPEG-1 teaches *transmission* of encoded video “*from a video encoding device to a corresponding video decoding device.*” Saber, ¶135.

d. Claim 4

MPEG-1 discloses or suggests claim 4's limitation. Saber, ¶136.

As described in §VII.A.2.a(v), MPEG-1's non_intra_quantizer_matrix indicates "*the default level of quantization*" in the sequence header. MPEG-1, 42, 43; Saber, ¶137. A coded video commences with a sequence header followed by one or more groups of pictures, and before each group of pictures, there may be another sequence header. MPEG-1, 32. MPEG-1 uses the non_intra_quantizer_matrix until the next occurrence of the sequence header. MPEG-1, 42, 43; Saber, ¶138. The non_intra_quantizer_matrix can then be redefined when another sequence header occurs in the bitstream (e.g., by specifying different matrix values) ("*the default level of quantization is updated*"). MPEG-1, 32; Saber, ¶138. Thus, the non_intra_quantizer_matrix can be updated by a subsequent sequence header ("*providing an indication of the updated default level of quantization*"). Saber, ¶138.

e. Claim 5

MPEG-1 discloses or suggests claim 5's limitation. Saber, ¶139.

As described in §VII.A.2.a(iv), MPEG-1 discloses a "*default level of quantization*" used to quantize the DCT coefficients in a video sequence ("*said sets of transform coefficient values*") until a next occurrence of a sequence header (e.g.,

as indicated by the `load_non_intra_quantizer_matrix` flag). MPEG-1, 42, 43; Saber, ¶140.

MPEG-1 further describes providing a quantizer scale (“`quantizer_scale`”) parameter in the slice header and (optionally) the macroblock header. MPEG-1, 39; Saber, ¶141. The quantizer scale is an unsigned integer between 1 and 31 that scales the reconstruction level of DCT coefficients until another quantizer scale is provided (e.g., in a subsequent slice or macroblock layer). MPEG-1, 45-46; Saber, ¶¶142-43. When the quantizer scale is set to “1,” it has no effect on the `non_intra_quantizer_matrix` set in the sequence header (“*the default level of quantization*”). Saber, ¶143. When the quantizer scale is set to an integer different from “1,” quantization is performed using the scaled `non_intra_quantizer_matrix` (“*a level of quantization different from the default level of quantization*”). Saber, ¶143.

MPEG-1 further discloses quantization of macroblocks in P-pictures depends on macroblock type (listed in Table D.19 below). MPEG-1, 110-11. Macroblock types “pred-mc” and “pred-cq” (red) include a quantizer scale (since “QUANT” is set to 1). MPEG-1, 110-11; Saber, ¶144. These two macroblock types apply to motion-compensated, coded non-intra macroblocks. MPEG-1, 112. In contrast, macroblock type “pred-mc” (blue) does not include quantizer scale (as

explained below, quantization is done using a previously established level).

MPEG-1, 110-12; Saber, ¶144.

Table D.19 -- Macroblock type VLC for P-pictures (table B.2b)

TYPE	VLC	INTRA	MOTION FORWARD	CODED PATTERN	QUANT
pred-mc	1	0	1	1	0
pred-c	01	0	0	1	0
pred-m	001	0	1	0	0
intra-d	0001 1	1	0	0	0
pred-mcq	0001 0	0	1	1	1
pred-cq	0000 1	0	0	1	1
intra-q	0000 01	1	0	0	1
skipped	N/A				

For P-pictures, non-intra macroblocks are quantized using the `non_intra_quantizer_matrix` (“niqmatrix”) and the quantizer scale (“quantscale”), regardless of whether the scale is provided at the slice or macroblock level.

MPEG-1, 115. This occurs using the equation

“ $\text{coeffqant} = (8 * \text{coefforig}) / (\text{quantscale} * \text{niqmatrix})$,” where “coeffqant” is the quantized coefficient and “coefforig” is the original DCT coefficient. MPEG-1,

115. As explained in §VII.A.1, MPEG-1 multiplies the non-intra quantizer

(specified via the sequence header) by the quantizer scale (specified at the slice or macroblock level) when reconstructing the DCT coefficients on the decoding side,

(“ $\text{dct_recon}[m][n] = (((2 * \text{dct_zz}[i] + \text{Sign}(\text{dct_zz}[i]) * \text{quantizer_scale} * \text{non_intra_quant}[m][n]) / 16)$ ”). MPEG-1, 53; Saber, ¶¶145-46.

Given that DCT coefficients of the non-intra macroblocks of P-pictures are quantized using non_intra_quantizer_matrix multiplied by the quantizer scale using the above equation, the quantized coefficients are quantized with a level of quantization “*different from the default level of quantization*” (i.e., the non_intra_quantizer_matrix). Saber, ¶¶146-47.

f. Claim 6

MPEG-1 discloses or suggests claim 6’s limitation. Saber, ¶148.

As explained in §VII.A.2.e, MPEG-1 quantizes coefficients with a level different from the non_intra_quantizer_matrix by disclosing the quantizer scale for the non_intra_quantizer_matrix. MPEG-1, 45. MPEG-1 provides the quantizer scale in the slice or macroblock header syntax. MPEG-1, 22, 46. The slice header includes data needed to decode a slice in a video frame. Saber, ¶149. An MPEG-1 decoder uses a given quantizer scale until it encounters another. MPEG-1, 45-46; Saber, ¶149. The quantizer scale is therefore an indication of a difference between a default quantization level and a scaled quantizer level provided to a decoding process (“*an indication of a difference between said level of quantization and the default level of quantization*”). Saber, ¶149.

A POSITA would have understood “*an indication of a difference between said level of quantization and the default level of quantization*” encompasses an indication that two levels of quantization are distinct or dissimilar. Saber, ¶150.

The quantizer scale that changes non_intra_quantizer_matrix used in encoding subsequent slices is therefore an indication of “*a difference*” between the non_intra_quantizer_matrix and the scaled non_intra_quantizer_matrix. Saber, ¶150.

If PO asserts this claim requires an arithmetic difference between “*said level of quantization and the default level of quantization*,” MPEG-1’s quantizer scale teaches that as well. Saber, ¶151. The ’148 Patent explains in some instances, “the actual level of inverse quantization is determined by adding a difference value to the default level of inverse quantization.” ’148Pat., 13:63-41:1. Saber, ¶¶152-53. As described in §VII.A.2.e, MPEG-1’s quantizer scale (“*indication of a difference*”) specifies the scale for non_intra_quantizer_matrix values (“*default level of quantization*”). Further, because multiplication is repeated addition, the quantizer scale indicates the number of times a value of the non_intra_quantizer_matrix is added to itself. Saber, ¶153. The quantizer scale thus indicates the number of times non_intra_quantizer_matrix values are added to themselves (that is, a number of arithmetic differences of the “*default level of quantization*”), which teaches “*an indication of a difference between said level of quantization and the default level of quantization*.” Saber, ¶153.

g. Claim 7

MPEG-1 discloses or suggests claim 7’s limitation. Saber, ¶154.

As described in §VII.A.2.e, MPEG-1's quantizer scale ("quantizer_scale") parameter is in the slice header syntax. MPEG-1, 39. When the quantizer scale is different from "1," the DCT coefficients are quantized at a different level than the default (as described in §VII.A.2.a(iv)) ("*an actual level of quantization . . . is different from the default level of quantization*"). MPEG-1, 42-43; Saber, ¶155.

MPEG-1's quantizer scale applies until another quantizer scale is specified. MPEG-1, 46. A POSITA would have understood there are scenarios in which the quantizer scale "*is adjusted from one frame of the digital video sequence to another such that the actual level of quantization applied to the sets of transform coefficients for a particular frame of the digital video sequence is different from the default level of quantization.*" Saber, ¶156.

As an example, MPEG-1 describes pictures consisting of "one or more slices." MPEG-1, 71. A picture may therefore include only a single slice. Saber, ¶157. In this scenario, the level of quantization is adjusted from one picture ("*frame*") to another by the quantizer scale parameter in the slice header syntax of the single slice forming the picture. Saber, ¶157.

h. Claim 8

MPEG-1 discloses or suggests claim 8's limitation. Saber, ¶158.

As described in §VII.A.2.f, the MPEG-1 quantizer scale indicates "*a difference*" with respect to the non_intra_quantizer_matrix (the "*default level of*

quantization,” discussed in §VII.A.2.a(iv) for 1[c]) and the quantization level applied as a result of scaling the default level with the quantizer scale parameter (*“said actual level of quantization”*). Saber, ¶159.

i. Claim 9

MPEG-1 discloses or suggests claim 9’s limitation. Saber, ¶160.

As described in §VII.A.2.f, the quantizer scale indicates *“a difference between”* the level of quantization indicated by the non_intra_quantizer_matrix (the *“default level of quantization,”* discussed in §VII.A.2.a(iv) for 1[c]) and the quantization level applied as a result of scaling the default level with the quantizer scale parameter (*“said actual level of quantization”*). Saber, ¶161. As further described in §VII.A.2.f (claim 6), this difference is indicated in the slice header that is *“provided to”* (and used by) the decoder for *“a decoding process”* until another quantizer scale is encountered either at the slice or macroblock layer. MPEG-1, 45-46; Saber, ¶161.

j. Claim 10

MPEG-1 discloses or suggests claim 10’s limitation. Saber, ¶162.

An MPEG-1 slice is a series of macroblocks, each including four luminance blocks and two chrominance blocks, where a block is an 8-pel-by-8-line section. MPEG-1, 20, 24, 26, 34. Both a macroblock and a slice correspond to *“a plurality of blocks of pixels ... grouped into one or more segments.”* Saber, ¶163.

MPEG-1's luminance and chrominance blocks are a *“plurality of blocks of pixels, into which a frame of said digital video sequence is divided”* for the reasons described in VII.A.2.a(i). Saber, ¶164. MPEG-1 also describes that each macroblock contains a 16-pel-by-16-line section of luminance (containing 4 arranged 8-pel-by-8-line luminance blocks) and spatially corresponding to 8-pel-by-8-line section of each chrominance component. MPEG-1, 34; Saber, ¶164. Each macroblock therefore discloses/suggests a *“segment”* into which *“a plurality of blocks of pixels”* (e.g., luminance blocks and chrominance blocks) *“are grouped.”* Saber, ¶164.

Further, as described in §VII.A.2.e, MPEG-1's quantizer scale is used with the non_intra_quantizer_matrix. MPEG-1, 115. The scale (*“quantizer_scale”*) is provided in the slice header syntax (MPEG-1, 39, 91) and/or in the macroblock level syntax (MPEG-1, 39, 72). Saber, ¶165. MPEG-1's non_intra_quantizer_matrix is thus scaled slice-by-slice or macroblock-by-macroblock, either of which being *“blocks of pixels...grouped into one or more segments.”* Saber, ¶165. MPEG-1's quantizer scale therefore discloses/suggests a *“level of quantization”* that is *“adjusted from one segment of a frame to another such that an actual level of quantization applied to the sets of transform coefficients for a particular segment of a frame is different from the default level of quantization.”* Saber, ¶165.

k. Claim 11

MPEG-1 discloses or suggests claim 11's limitation. Saber, ¶166.

As described in §VII.A.2.j, the quantizer scale is provided in the slice header or in the macroblock header syntax to indicate which level of quantization is used. MPEG-1, 39, 91. As described in §VII.A.2.f, the quantizer scale is “*an indication of a difference between...the default level of quantization*” (i.e., the non_intra_quantizer_matrix) and the quantization level applied after scaling the default level with the quantizer scale parameter (“*said level of quantization*”). Saber, ¶167.

3. Claim 12 and Dependents

a. Claim 12

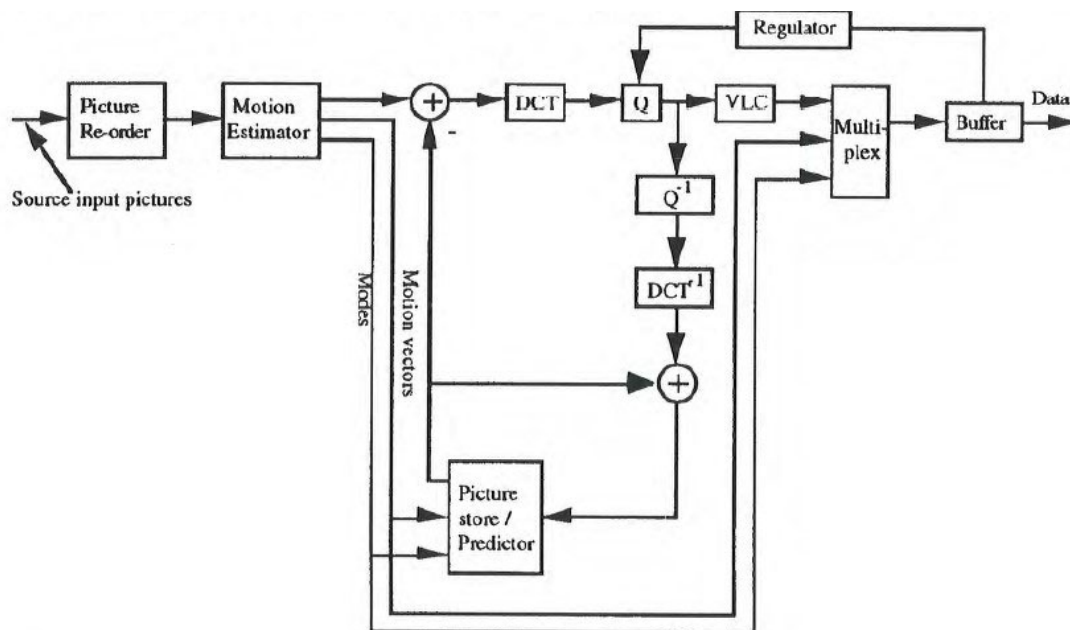
i. 12[pre]

If limiting, MPEG-1 discloses or suggests 12[pre]. Saber, ¶168.

As described in §VII.A.2.a(i), MPEG-1 discloses a method “*to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels.*” Saber, ¶169.

Further, Figure 2 of MPEG-1 (reproduced below) describes an encoder and its operations to produce “a valid coded bitstream” that can be used “for

compressing video sequences to bitrates around 1,5 Mbit/s.” MPEG-1, 12, 14, 22; Saber, ¶¶170-71. The figure shows the “main functional blocks” of an encoder. MPEG-1, 14. MPEG-1 thus discloses/suggests “a video encoder for encoding a digital video sequence.” Saber, ¶171.



where

DCT is discrete cosine transform
DCT⁻¹ is inverse discrete cosine transform
Q is quantization
Q⁻¹ is dequantization
VLC is variable length coding

Figure 2 -- Simplified video encoder block diagram

ii. 12[a]

MPEG-1 discloses or suggests 12[a]. Saber, ¶172.

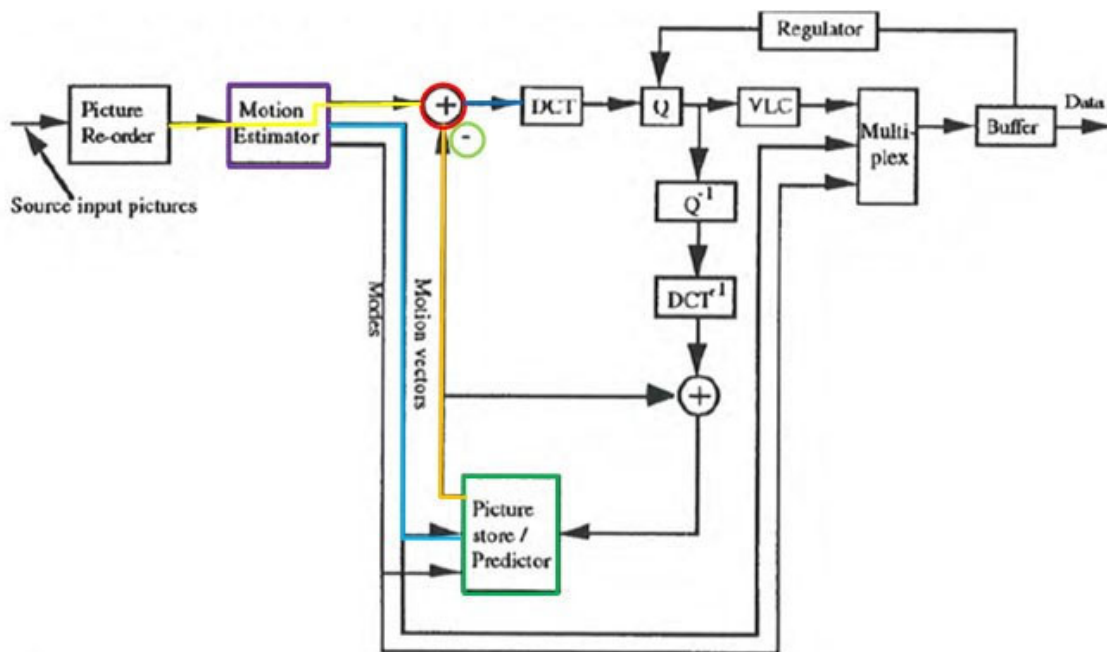
As described in §VI.A.1, the recited function of this means-plus-function limitation is “applying motion compensated prediction to blocks of pixels” and “provid[ing] corresponding blocks of prediction error values,” and the

corresponding structure is a motion compensated predictor and combiner. '148Pat., 7:66-8:4 (motion compensated predictor), 8:5-9 (combiner); Saber, ¶173.

As described in §VII.A.2.a(ii), MPEG-1 discloses “*applying motion compensated prediction to blocks of pixels.*” Figure 2 (annotated below) illustrates that the video encoder includes a “Motion Estimator” (purple) that generates motion vectors (light blue) provided to a “Picture store / Predictor” (green). MPEG-1, 14; Saber, ¶174. Referring to Figure 2, MPEG-1 describes that “the encoder estimates motion vectors for each 16 by 16 macroblock in the picture.” MPEG-1, 14; Saber, ¶¶175-76. A POSITA would have understood that the “Picture store / Predictor” that forms predictions (orange) based on the generated motion vectors for each macroblock is a motion compensated predictor that applies “*motion compensated prediction to blocks of pixels*” in the same manner as the motion compensated predictor in the '148 Patent. Saber, ¶176.

MPEG-1 also discloses/suggests “*provid[ing] corresponding blocks of prediction error values*” for the reasons discussed in §VII.A.2.a(ii). Saber, ¶177. MPEG-1's Figure 2 illustrates that the difference (red) of the video encoder takes the difference (illustrated with “-” sign, light green) between input pictures (yellow) and predictions formed by the “Picture store / Predictor” (orange). Saber, ¶177. A POSITA would have known that this difference (blue) is “*blocks of prediction error values.*” Saber, ¶177. Further, the '148 Patent explains that a

combiner subtracts predicted pixel values from input pixel values. '148Pat., 8:5-9. The difference block (red) similarly subtracts predicted pixel values (the result of motion-compensated prediction) from input pixels (input pictures) and operates in the same manner as the combiner described by the '148 Patent. Saber, ¶¶178-79. A POSITA therefore would have known that the difference block (red) in a MPEG-1 encoder that provides the “*blocks of prediction error values*” is a combiner that operates in the same manner as the combiner in the '148 Patent. Saber, ¶179.



where

DCT is discrete cosine transform
DCT⁻¹ is inverse discrete cosine transform
Q is quantization
Q⁻¹ is dequantization
VLC is variable length coding

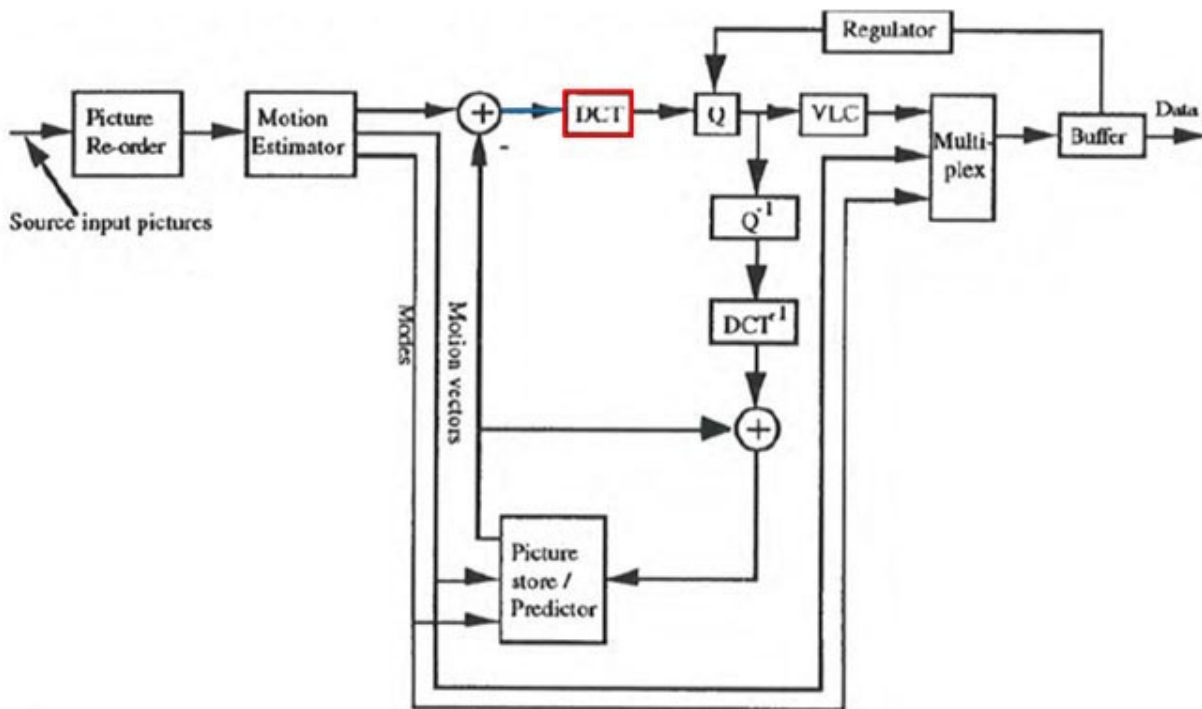
Figure 2 -- Simplified video encoder block diagram

iii. 12[b]

MPEG-1 discloses or suggests 12[b]. Saber, ¶180.

As described in §VI.A.2, the recited function of this means-plus-function limitation is “*providing sets of transform coefficient values representative of said blocks of prediction error values*,” and the corresponding structure is the disclosed transform block. ’148Pat., 18:66-19:3; Saber, ¶181.

MPEG-1 discloses/suggests “*transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values*” for the reasons discussed in §VII.A.2.a(iii). Saber, ¶182. Further, the encoder illustrated in Figure 2 of MPEG-1 (annotated below) includes a DCT block (red) that receives prediction error signals (blue) from the combiner. MPEG-1, 14-15; Saber, ¶183. Referencing the encoder of Figure 2, MPEG-1 describes that the “error signal is separated into 8 by 8 blocks (4 luminance and 2 chrominance blocks in each macroblock) and a [DCT] is performed on each block” to produce DCT coefficients. MPEG-1, 15; Saber, ¶¶183-85. A POSITA would have understood that the illustrated DCT block, which receives the prediction error signals and generates the input to the quantizer (DCT coefficients), “provid[es] sets of transform coefficient values” from the received prediction error signals in the same manner as the transform block in the ’148 Patent. Saber, ¶185.



where

DCT is discrete cosine transform
DCT⁻¹ is inverse discrete cosine transform
Q is quantization
Q⁻¹ is dequantization
VLC is variable length coding

Figure 2 -- Simplified video encoder block diagram

iv. 12[c]

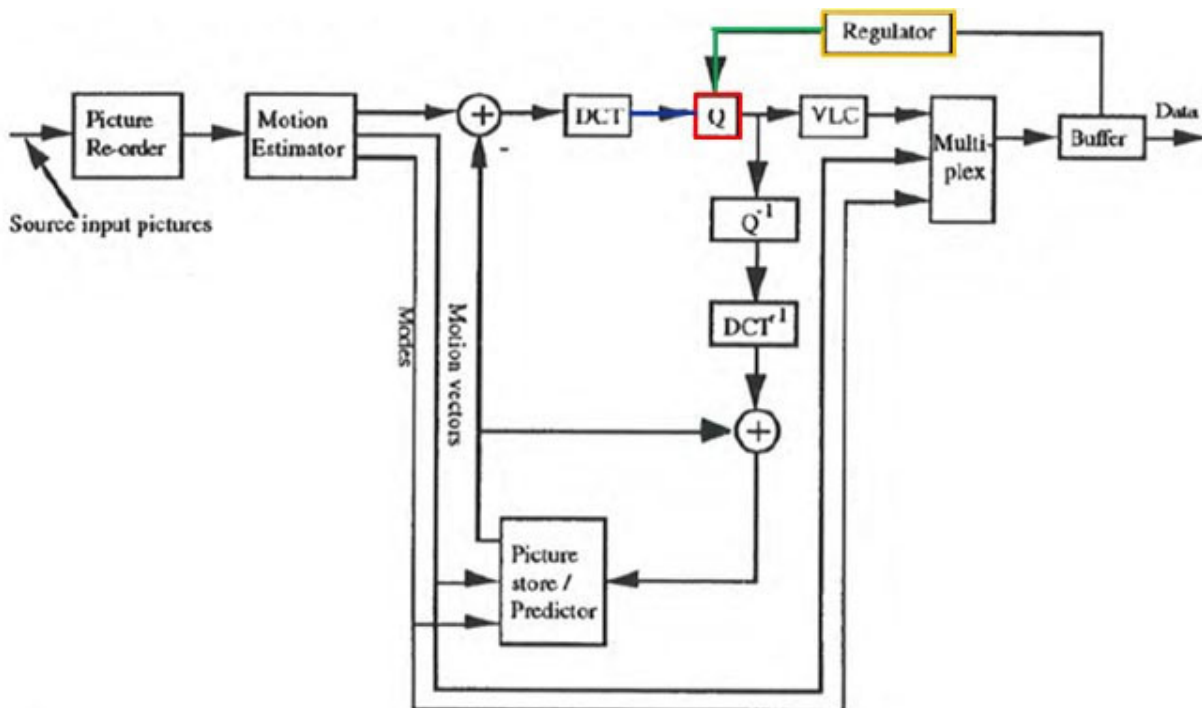
MPEG-1 discloses or suggests 12[c]. Saber, ¶186.

As described in §VI.A.3, the recited function of this means-plus-function limitation is “*selecting a default level of quantization for quantizing said sets of transform coefficient values,*” and the corresponding structure is the disclosed control manager. ’148Pat., 6:16-22; 19:60-65; Saber, ¶187.

MPEG-1 discloses/suggests “*selecting a default level of quantization for quantizing said sets of transform coefficient values*” for the same reasons as described in §VII.A.2.a(iv). Saber, ¶188. MPEG-1’s video encoder (annotated below) includes a quantization block (labeled “Q,” red) that quantizes the blocks of incoming DCT coefficients (blue), converting them to quantized DCT coefficients. MPEG-1, 14–15; Saber, ¶188. Referencing the encoder of Figure 2, MPEG-1 describes that because the overall data rate of a video is variable (due to different picture types and variable length coding) when a fixed-rate channel is used “[t]he status of [a] buffer may be monitored to control the number of bits generated by the encoder. Controlling the quantization process is the most direct way of controlling the bitrate.” MPEG-1, 15; Saber, ¶¶189-90.

The encoder illustrated in MPEG-1 also includes a “Regulator” block (yellow) that receives information from a “Buffer” block and provides information (green) to the quantization block. MPEG-1, 14; Saber, ¶191. Further, the Regulator block of the illustrated MPEG-1 encoder is the only block that exchanges data with both the quantization block and the Buffer block. MPEG-1, 14; Saber, ¶191. A POSITA would have understood that the Regulator block therefore performs the described “[c]ontrolling [of] the quantization process” to control the bitrate. Saber, ¶191. Further, to control the quantization process, the Regulator must *select* the quantization used. Saber, ¶191. A POSITA would therefore have understood that

the Regulator block of the MPEG-1 encoder is used for “*selecting a default level of quantization*” (e.g., based on the information from the buffer) “*for quantizing*” the transform coefficient values by the quantization block. Saber, ¶191. The Regulator block of the MPEG-1 operates in the same manner as the control manager in the ’148 Patent. Saber, ¶191.



where

DCT is discrete cosine transform
DCT⁻¹ is inverse discrete cosine transform
Q is quantization
Q⁻¹ is dequantization
VLC is variable length coding

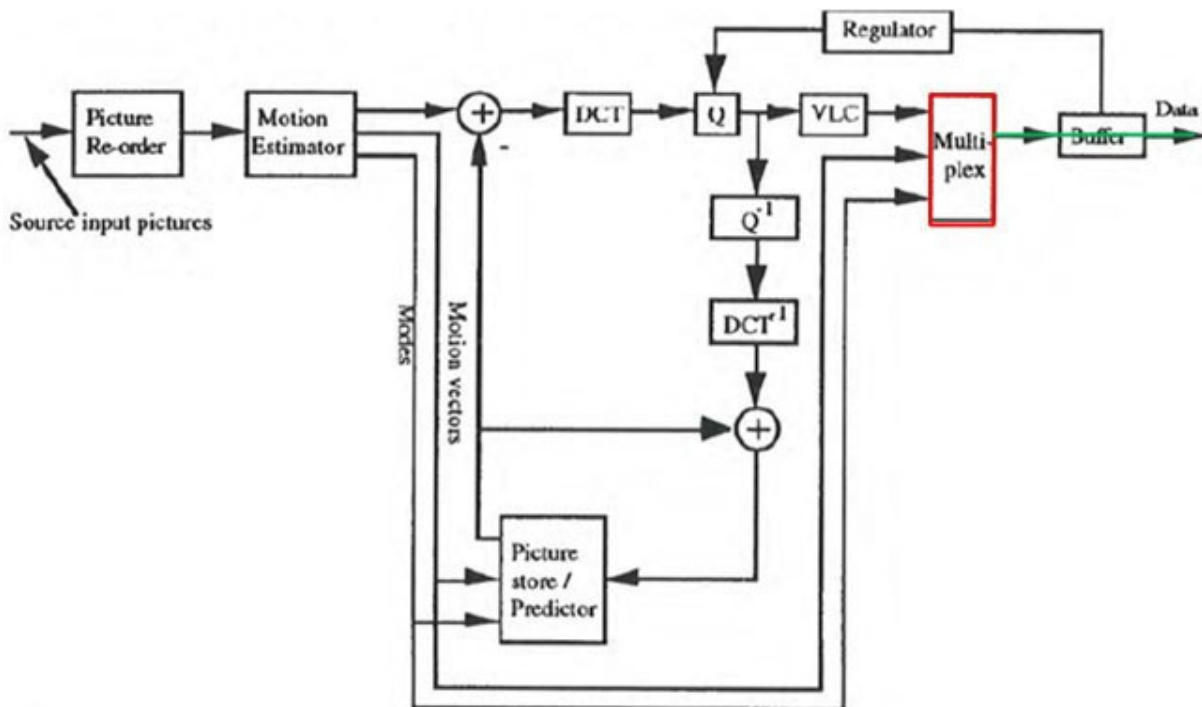
Figure 2 -- Simplified video encoder block diagram

v. 12[d]

MPEG-1 discloses or suggests 12[d]. Saber, ¶192.

As described in §VI.A.4, the recited function of this means-plus-function limitation is “*providing an indication of said default level of quantization in the encoded bit-stream,*” and the corresponding structure is the video multiplex coder. ’148Pat., 6:38-47; 19:60-65; Saber, ¶193.

MPEG-1 discloses/suggests “*providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons as described in §VII.A.2.a(v). Saber, ¶194. The encoder illustrated in Figure 2 of MPEG-1 (annotated below) includes a multiplex block (red). MPEG-1, 14. The multiplex block is illustrated to receive processed data from other blocks of the encoder, including the quantization block, and to output a coded bitstream (green). Saber, ¶195. A POSITA would therefore have understood that the multiplex block of the MPEG-1 encoder provides an encoded bitstream “representative of the digital video sequence” to a decoder in the same manner as the video multiplex coder in the ’148 Patent. MPEG-1, 14-15; Saber, ¶195. The encoded bitstream includes the sequence level header that indicates the “default level of quantization,” as described in §VII.A.2.a(v). Saber, ¶195.



where

DCT is discrete cosine transform

DCT^{-1} is inverse discrete cosine transform

Q is quantization

Q^{-1} is dequantization

VLC is variable length coding

Figure 2 -- Simplified video encoder block diagram

b. Claim 13

MPEG-1 discloses or suggests claim 13's limitation. Saber, ¶196.

MPEG-1 discloses/suggests “wherein said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from the default level of quantization” as described in

§VII.A.2.e, and because its encoder (“video encoder”) conforms to its encoding method. Saber, ¶197.

c. Claim 14

MPEG-1 discloses or suggests claim 14’s limitation. Saber, ¶198.

MPEG-1 discloses/suggests “*arranged to provide an indication of a difference between said level of quantization and the default level of quantization*” for the reasons described in §VII.A.2.f. Saber, ¶199. A POSITA would have understood that MPEG-1’s disclosed encoder (“*a video encoder*”) performs MPEG-1’s encoding method that provides the indication “*in the encoded bit-stream*” sent to a decoder. Saber, ¶200.

d. Claim 15

MPEG-1 discloses or suggests claim 15’s limitation. Saber, ¶201.

MPEG-1 discloses/suggests “*wherein the default level of quantization can be updated and the video encoder is arranged to provide an indication of the updated default level of quantization*” for the reasons described in §VII.A.2.d. Saber, ¶202. A POSITA would have understood that MPEG-1’s disclosed encoder (“*a video encoder*”) conforms to MPEG-1’s encoding method that provides the indication “*in the encoded bit-stream*” sent to a decoder. Saber, ¶203.

e. Claim 16

MPEG-1 discloses or suggests claim 16’s limitation. Saber, ¶204.

As discussed in §0, MPEG-1 discloses a video encoder in which “*said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from*” (e.g., based on quantizer_scale) quantization specified by the non_intra_quantizer_matrix. Saber, ¶205. A POSITA would have understood that quantizer_scale, when set to a value different from “1,” provides “*a level of quantization different from the updated default level of quantization*” (e.g., non_intra_quantizer_matrix). Saber, ¶205.

f. Claim 17

MPEG-1 discloses or suggests claim 17’s limitation. Saber, ¶206.

MPEG-1 describes a coded representation that can be used for compressing video sequences to bitrates around 1.5 Mbit/s and was developed to operate from storage media offering a continuous transfer rate of about 1.5 Mbit/s. MPEG-1, 12; Saber, ¶207. The coded representation can be used for television, workstations, and personal computer displays. MPEG-1, 12; Saber, ¶207.

The ’148 Patent explains that “*a multimedia terminal*” includes elements that capture, encode, and multiplex multimedia data streams for transmission via a communication network; and elements that receive, de-multiplex, decode, and display received multimedia content. ’148Pat., 22:56-65; Saber, ¶¶208-09. The ’148 Patent further explains that a “*multimedia*” terminal can be used for

applications such as the retrieval and/or streaming of video clips (e.g., from a content server). '148Pat., 23:1-5; Saber, ¶¶208-09.

A POSITA would understand that televisions and personal computer displays can, for example, include elements that receive, de-multiplex, decode, and display multimedia content (e.g., audio and video data). Saber, ¶210. A POSITA would have further understood that workstations can include elements that capture, encode, and multiplex multimedia data streams to be transmitted, as well as decoding and displaying the same (e.g., video from a content server). Saber, ¶210. In other words, MPEG-1 describes that it may be utilized in systems (e.g., televisions, workstations, and personal computer displays) that perform the functions of a “*multimedia terminal*.” Saber, ¶210. MPEG-1 therefore discloses/suggests that MPEG-1 can be implemented in systems including “*a multimedia terminal*.” Saber, ¶210.

g. Claim 18

MPEG-1 discloses or suggests claim 18's limitation. Saber, ¶211.

As described in §VII.A.3.f, MPEG-1 describes that the coded representation of MPEG-1 can be used for television, workstations, and personal computer displays. MPEG-1, 12; Saber, ¶212. MPEG-1 further describes that motion video encoded using MPEG-1 can be transmitted and received over existing and future networks. MPEG-1, 12. A POSITA would have known that an encoded MPEG-1

data stream could be transmitted wirelessly over a radio channel (e.g., to a television or personal computer) via “*a radio telecommunications device.*” Saber, ¶212.

h. Claim 19

MPEG-1 discloses or suggests claim 19’s limitation. Saber, ¶213.

MPEG-1 discloses/suggests “*wherein said level of quantization applied to said sets of transform coefficient values can be adjusted from one frame of the digital video sequence to another so as to apply an actual level of quantization to the sets of transform coefficients for a particular frame of the digital video sequence and wherein said actual level of quantization is different from the default level of quantization*” for the reasons described in §VII.A.2.g, and because MPEG-1’s encoder (“*a video encoder*”) performs MPEG-1’s encoding method. Saber, ¶¶214-15.

i. Claim 20

MPEG-1 discloses or suggests claim 20’s limitation. Saber, ¶216.

MPEG-1 discloses/suggests “*provid[ing] an indication of said difference with respect to the default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons described in §VII.A.2.i, and because MPEG-1’s encoder (“*a video encoder*”) performs MPEG-1’s encoding method. Saber, ¶¶217-18.

j. Claim 21

MPEG-1 discloses or suggests claim 21's limitation. Saber, ¶219.

As reflected in the table below, each of the elements of claim 21 is substantially similar to an element of claims 10 or 11 (both of which are Challenged Claims, and discussed in §VII.A.2.j and §VII.A.2.k, respectively). Saber, ¶220. For example, claim 21 recites “*arranged to group the plurality of blocks...divided into one or more segments,*” and claim 10 recites an encoding method in which “*the plurality of blocks of pixels...are grouped into one or more segments.*” As a further example, claim 21 recites “*adjust[ing] the level of quantization...from one segment of a frame to another,*” and claim 10 recites an encoding method in which “*the level of quantization...is adjusted from one segment of a frame to another.*” MPEG-1 therefore discloses/suggests claim 21's limitation for the reasons discussed in §VII.A.2.j and §VII.A.2.k. A POSITA would have understood that MPEG-1's encoder (“*a video encoder*”) performs MPEG-1's encoding method. Saber, ¶220.

Claim 21	Corresponding Limitation from Claim 10/11
21[a]	wherein the plurality of blocks of pixels, into which a frame of said digital video sequence is divided, are grouped into one or more segments; (§VII.A.2.j)
21[b]	wherein the level of quantization applied to said sets of transform coefficient values is adjusted from one segment of a frame to another; (§VII.A.2.j)

Claim 21	Corresponding Limitation from Claim 10/11
21[c]	such that an actual level of quantization applied to the sets of transform coefficients for a particular segment of a frame is different from the default level of quantization; (§VII.A.2.j)
21[d]	providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization. (§VII.A.2.k)

k. Claim 22

MPEG-1 discloses or suggests claim 22's limitation. Saber, ¶221.

MPEG-1 discloses/suggests “*provid[ing] an indication of said difference with respect to the default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons described in §VII.A.2.k, and because MPEG-1's encoder (“*a video encoder*”) performs MPEG-1's encoding method. Saber, ¶¶222-23.

4. Claim 23

a. 23[pre]

If limiting, MPEG-1 discloses/suggests 23[pre] for substantially the same reasons as 12[pre]. §VII.A.3.a(i); Saber, ¶224.

b. 23[a]

MPEG-1 discloses/suggests 23[a] for substantially the same reasons discussed in §VII.A.3.a(ii) (12[a]) and §VI.A.1. Saber, ¶¶225-26.

c. 23[b]

MPEG-1 discloses/suggests 23[b] for substantially the same reasons discussed in §VII.A.3.a(iii) (12[b]) and §VI.A.2. Saber, ¶¶227-28.

d. 23[c]

MPEG-1 discloses or suggests 23[c]. Saber, ¶229.

MPEG-1 discloses/suggests “*selecting a default level of quantization for quantizing said sets of transform coefficient values,*” part of the function of the “*module for selecting,*” and the corresponding structure of control manager for the reasons discussed in §VII.A.3.a(iv) (12[c]) and §VI.A.3. Saber, ¶230.

Additionally, 23[c] recites “*wherein the default level is specific to the video coding applications.*” Saber, ¶231. The sole disclosure in the ’148 patent of an application-specific default level is an embodiment where “a sequence QP is not transmitted but an application specific constant is used as sequence QP instead.” ’148Pat, 25:25-27. In other words, the default level of quantization is consistent for video sequences processed by a video application. Saber, ¶¶231-32.

As explained in §VII.A.2.a(iv), MPEG-1 describes that when the flag `load_non_intra_quantizer_matrix` is set to “0,” the `non_intra_quantizer_matrix` is “reset” to “default values” of all 16s, alternatively referred to a “flat default quantization matrix.” MPEG-1, 42, 83-84; Saber, ¶233. When `load_non_intra_quantizer_matrix` is set to 0, values for the

non_intra_quantizer_matrix are not transmitted. MPEG-1, 42; Saber, ¶233. So when an MPEG-1 video sequence does not specify any non_intra_quantizer_matrix values (i.e., load_non_intra_quantizer_matrix is not set to 1 during the sequence and non_intra_quantizer_matrix values are not transmitted), the flat default quantization matrix operates at a “*default level*” specified by a video coding standard (e.g., MPEG-1). Saber, ¶233.

The “*default level*” (of the flat default quantization matrix) may be used by any “*video coding application*” that implements MPEG-1 encoding. Saber, ¶234. Thus, when a “*video coding application*” that implements MPEG-1 does not set load_non_intra_quantizer_matrix to 1 during a video sequence, the flat default quantization matrix operates as a “*default level*” that is “*specific to the video coding applications*” in a manner analogous to the disclosed embodiment in the ’148 patent. Saber, ¶234; ’148Pat, 25:25-27.

B. Ground 2: Obviousness over H.263

1. H.263

H.263 (EX-1006) (“Video coding for low bit rate communication”) is a video compression standard that “specifies a coded representation” for compressing the moving picture component of video. H.263, i; Saber, ¶¶235-36. The revision of H.263 referenced herein (EX-1006) was ratified by the ITU-T in

February 1998 and was publicly available by May 11, 2000, making it prior art under §102(b). H.263, i; Hall-Ellis, 46-53.

H.263 defines a “source format” for video sequence pictures, which are coded as luminance (representing brightness) and color difference (also called “chrominance”) components. H.263, 7; Saber, ¶237. H.263 describes “a hierarchical structure with four primary layers” for the encoded video syntax: (1) picture; (2) Group of Blocks (GOB), or slice, or video picture segment; (3) macroblock; and (4) block (each block representing luminance or chrominance). H.263, 11, 13; Saber, ¶238.

H.263 describes INTER coding (a “P-picture”) where a picture and/or macroblocks are predicted using a prior reference picture, and INTRA coding (an “I-picture”) where no temporal prediction is applied. H.263, 11; Saber, ¶239. When combining INTER coding with motion compensation, the decoder will, by default, accept one motion vector per macroblock. H.263, 11-12; Saber, ¶239.

H.263 encodes macroblocks by applying a DCT to the blocks in each macroblock, generating transform coefficients. H.263, 9, 11, 47, Fig. 3; Saber, ¶240. For INTRA blocks, the DCT transform is performed on the luminance and chrominance blocks. H.263, 47; Saber, ¶¶240-41. For INTER blocks, the DCT transform is performed on the prediction error (i.e., the difference between the

values of the luminance and chrominance blocks, and their respective predictions).

H.263, 47-48; Saber, ¶¶240-41.

Transform coefficients are quantized at the encoder and are transmitted (to a decoder) after further encoding. H.263, 9, 11-13, 46, Fig. 3; Saber, ¶242.

Quantization uses a quantizer value (labeled “QUANT”) that is specified by information provided at the picture layer (“PQUANT”), and optionally at the GOB layer (“GQUANT”) and/or macroblock layer (“DQUANT”) within each picture.

H.263, 12-13, 30, 32-33, 36-37, 46; Saber, ¶242.

2. Claim 1 and Dependents

a. Claim 1

i. 1[pre]

If limiting, H.263 discloses or suggests it. Saber, ¶243.

H.263 “specifies a coded representation” for “compressing the moving picture component of” video (“*a method of encoding a digital video sequence*”).

H.263, i.

And H.263 explains that a “video coder provides a self-contained digital bitstream” (“*for use in a video coding application to produce an encoded video bitstream representative of the digital video sequence*”). H.263, 2; Saber, ¶244.

H.263 operates on non-interlaced pictures (*frames*) occurring at a Picture Clock Frequency (PCF) (e.g., approximately 29.97 times per second). H.263, 7;

Saber, ¶245. This is a “*digital video sequence comprising a number of frames.*”

Saber, ¶245.

In H.263, pictures are coded as luminance (Y) and color difference (C_B and C_R) components, and “picture size is measured as the picture width times height for the [Y] component, measured in pixels.” H.263, 6, 7. H.263 therefore discloses/suggests “*each frame of said sequence comprising an array of pixels.*” Saber, ¶246.

Each picture is divided into GOBs, and each GOB is divided into macroblocks. H.263, 9, 11. Each macroblock “consists of four luminance blocks and the two spatially corresponding colour difference blocks,” where each luminance and chrominance block (each a *block*) “relates to 8 pixel by 8 lines of Y, C_B, or C_R” (“*each frame ... divided into a plurality of blocks, each block comprising a certain number of said pixels*”). H.263, 11; Saber, ¶¶247-48.

ii. 1[a]

H.263 discloses or suggests 1[a]. Saber, ¶249.

As described in §VII.B.2.a(i), each macroblock consists of four luminance blocks and two chrominance blocks (each a “*block of pixels*”). Saber, ¶250. In H.263’s INTER coding mode, INTER macroblocks within a picture (“*a frame*”) are augmented by motion compensation, and each macroblock is encoded with a motion vector that indicates the spatial relationship between the pixels being

predicted (corresponding to the macroblock) and the pixels in the reference picture (from which the prediction is formed). H.263, 11-12, 38; Saber, ¶251. Each macroblock's "motion vector is used for all pixels in all four luminance blocks in the macroblock," and the motion vector is also used for both chrominance blocks of the macroblock. H.263, 45. This corresponds to "*encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels.*" Saber, ¶¶252-53.

H.263 video encoding uses "[a] hybrid of inter-picture prediction to utilize temporal redundancy and transform coding of the remaining signal to reduce spatial redundancy." H.263, 2. The "remaining signal" is a "*prediction error.*" Saber, ¶254. When reconstructing each luminance and chrominance block of an INTER macroblock in H.263 (e.g., during a decoding process), "the reconstruction is formed by summing the prediction and the result of the inverse transformation. The summation is performed on a pixel basis." H.263, 47-48. The result of the inverse transformation is described as "reconstructed prediction error." H.263, 48; Saber, ¶255. In this manner, H.263 reconstructs "*prediction error values*" when reconstructing "*blocks of pixels.*" Saber, ¶255.

H.263 also illustrates this step. The encoder's "main elements are prediction, block transformation and quantization." H.263, 9. In Figure 3 (annotated below) of H.263, the encoder generates (red) the difference between the video input (orange)

and the result of motion-compensated prediction (green). This difference (blue) corresponds to the prediction error. Saber, ¶256. H.263 thus discloses/suggests “producing corresponding blocks of prediction error values” based on the applied motion-compensated prediction. Saber, ¶257.

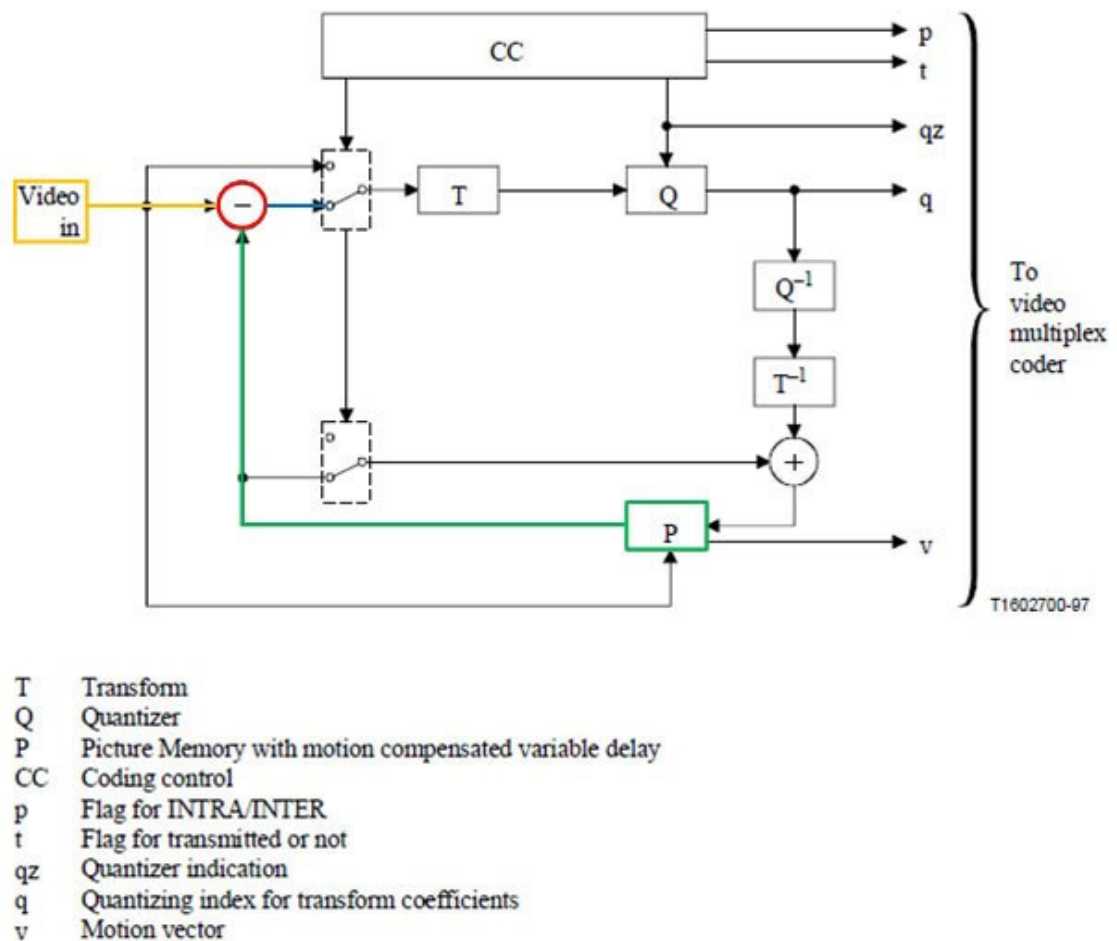
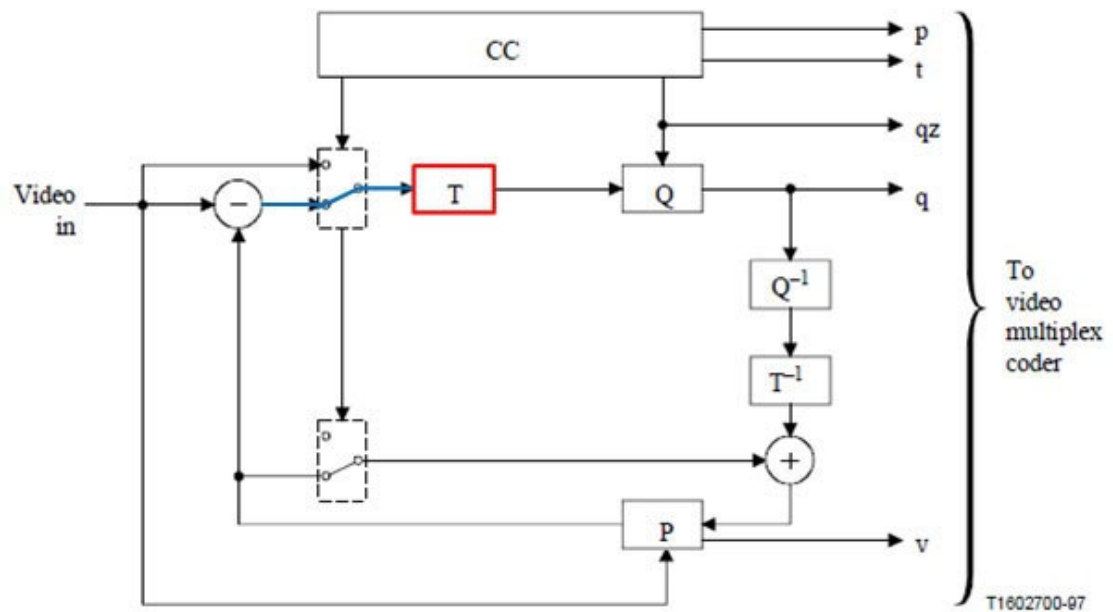


Figure 3/H.263 – Source coder

iii. 1[b]

H.263 discloses or suggests 1[b]. Saber, ¶258.

H.263 utilizes “transform coding of the remaining signal”—though, for example, DCT (“*applying a transform coding technique*”)—“to reduce spatial redundancy.” H.263, 2, 47; Saber, ¶259. And as described in §VII.B.2.a(ii), the “remaining signal” is a “*prediction error*,” and H.263 produces “*prediction error values*” for each luminance and chrominance block of an INTER macroblock (“*to said blocks*”). Saber, ¶260. This is illustrated in H.263’s Figure 3 (annotated below), where INTER coding applies a transform (red) to the prediction error values (blue). Saber, ¶261.



T	Transform
Q	Quantizer
P	Picture Memory with motion compensated variable delay
CC	Coding control
p	Flag for INTRA/INTER
t	Flag for transmitted or not
qz	Quantizer indication
q	Quantizing index for transform coefficients
v	Motion vector

Figure 3/H.263 – Source coder

The “[t]ransmitted blocks are transformed and resulting coefficients are quantized and entropy coded.” H.263, 11. H.263 thus discloses or suggests applying the coding technique (e.g., DCT) to “*produce sets of transform coefficient values representative of said blocks of prediction error values.*” Saber, ¶262.

iv. 1[c]

H.263 discloses or suggests 1[c]. Saber, ¶263.

In co-pending litigation, PO has pointed to H.264 parameters “pic_init_qp_minus26,” “slice_qp_delta,” and “mb_qp_delta” as examples of “*defining a default level of quantization for use in encoding of the digital video sequence.*” Nokia Complaint, 415-16, 424-30. Those parameters set quantization information at the picture, slice, and macroblock layers, respectively. H.264, 32, 36, 41; Saber, ¶¶264-67. Thus, the highest-layer quantization parameter that PO points to in H.264 is in the picture layer. Saber, ¶268. H.264 does not specify quantization information at higher layers (e.g., the sequence layer). H.264, 31; Saber, ¶269. To the extent PO argues that a picture-layer quantization parameter in H.264 meets this limitation,⁹ H.263’s quantization parameter—PQUANT (described below)—likewise meets it under the same interpretation. Saber, ¶270.

H.263 describes that “[t]ransmitted blocks are transformed and resulting coefficients are quantized and entropy coded.” H.263, 11. H.263 further describes that when performing inverse quantization (e.g., during a decoding process), coefficients are reconstructed according to the following formulas:

$$\begin{aligned} |\text{REC}| &= \text{QUANT} \cdot (2 \cdot |\text{LEVEL}| + 1) && \text{if QUANT} = \text{"odd"} \\ |\text{REC}| &= \text{QUANT} \cdot (2 \cdot |\text{LEVEL}| + 1) - 1 && \text{if QUANT} = \text{"even"} \end{aligned}$$

⁹ Petitioner reserves the right to argue other constructions in other proceedings.

H.263, 46. LEVEL refers to a coded (i.e., quantized) coefficient, and QUANT refers to the quantizer used. H.263, 30, 41; Saber, ¶¶271-73. H.263 references the inverse quantization formula, used during a decoding process, in describing the quantizer operation of encoding. H.263, 13. Thus, H.263 discloses or suggests a quantizer (QUANT) used during encoding. Saber, ¶273. H.263 further describes that there are 31 possible quantizers that can be used during encoding (other than for the first coefficient of INTRA blocks). H.263, 12; Saber, ¶274. Since H.263 describes quantization during encoding, and further describes 31 possible quantizers, H.263 discloses/suggests “*defining*” a quantizer during encoding. Saber, ¶274.

H.263 provides quantizer information (“PQUANT”) in the picture layer. §VII.B.1; H.263, 30, Fig. 6. PQUANT is “[a] fixed length codeword of 5 bits which indicates the quantizer QUANT to be used for the picture until updated by any subsequent GQUANT or DQUANT. The codewords are the natural binary representations of the values of QUANT which, being half the step sizes, range from 1 to 31.” H.263, 30. PQUANT thus indicates which quantizer to use for a picture. Saber, ¶¶275-76.

“Data for each picture consists of a picture header,” of which PQUANT is a part, meaning that every picture includes PQUANT. H.263, 22, Fig. 7; Saber, ¶¶277-78. In contrast, neither GQUANT nor DQUANT are necessarily included at

the GOB or macroblock layers within a picture. Saber, ¶¶279-82. GQUANT is part of a GOB header, which is not transmitted (for the first GOB in a picture) or may be empty (for all other GOBs in the picture). H.263, 32. And DQUANT is only included for certain macroblock types. H.263, 33, 36, Table 9. PQUANT thus defines the quantizer (QUANT) for a picture and may be updated by GQUANT or DQUANT parameters specified for GOBs or macroblocks, respectively, within the picture. Saber, ¶283. If no GOBs or macroblocks subsequently update QUANT using GQUANT or DQUANT, however, PQUANT will be used for quantizing the entirety of the picture. Saber, ¶283. PQUANT thus “*defin[es] a default level of quantization.*” Saber, ¶283. A POSITA would understand this to render obvious the use of a sequence-level parameter. Saber, ¶283. To the extent PO argues that a picture-layer quantization parameter (e.g., `pic_init_qp_minus26` of H.264) is “*for use in encoding of the digital video sequence,*” PQUANT in H.263 meets this limitation under the same interpretation. Saber, ¶284.

H.263 further describes that “[w]ithin a macroblock the same quantizer is used for all coefficients except the first one of INTRA blocks.” H.263, 12-13. In other words, the same quantizer (QUANT) is used for all the coefficients associated with the luminance and chrominance blocks that make up an INTER macroblock. Saber, ¶285. H.263 therefore discloses/suggests a default level of quantization (e.g., QUANT as set by PQUANT) used to “*quantize the sets of*

transform coefficient values” (the coefficients associated with the blocks of an INTER macroblock). Saber, ¶285.

v. 1[d]

H.263 discloses or suggests 1[d]. Saber, ¶286.

As described in VII.B.2.a(iv), H.263 discloses PQUANT (“*default level of quantization*”). H.263 additionally explains that PQUANT is encoded as part of a picture header. H.263, 22, Fig. 7. H.263 therefore discloses/suggests “*providing an indication of the default level of quantization.*” Saber, ¶287.

H.263 describes performing inverse quantization, as part of a decoding process, using quantizer QUANT. H.263, 44, 46. And as described in §VII.B.2.a(iv), H.263 uses PQUANT to set QUANT during decoding. H.263 thus discloses or suggests providing PQUANT “*to a decoding process.*” Saber, ¶288.

b. Claim 2

H.263 discloses or suggests claim 2’s limitation. Saber, ¶289.

As described in §VII.B.2.a(v), H.263 discloses that PQUANT (“*said indication of the default level of quantization*”) is provided in a picture header. H.263 describes that “[t]he video coder provides a self-contained digital bitstream.”

H.263, 2. This is illustrated in Figure 6, annotated below, in which picture layer data includes PQUANT (red) in a “video bitstream.” Saber, ¶¶290-91. H.263

therefore discloses/suggests that PQUANT is “*provided in the encoded bit-stream representative of the digital video sequence.*” Saber, ¶291.

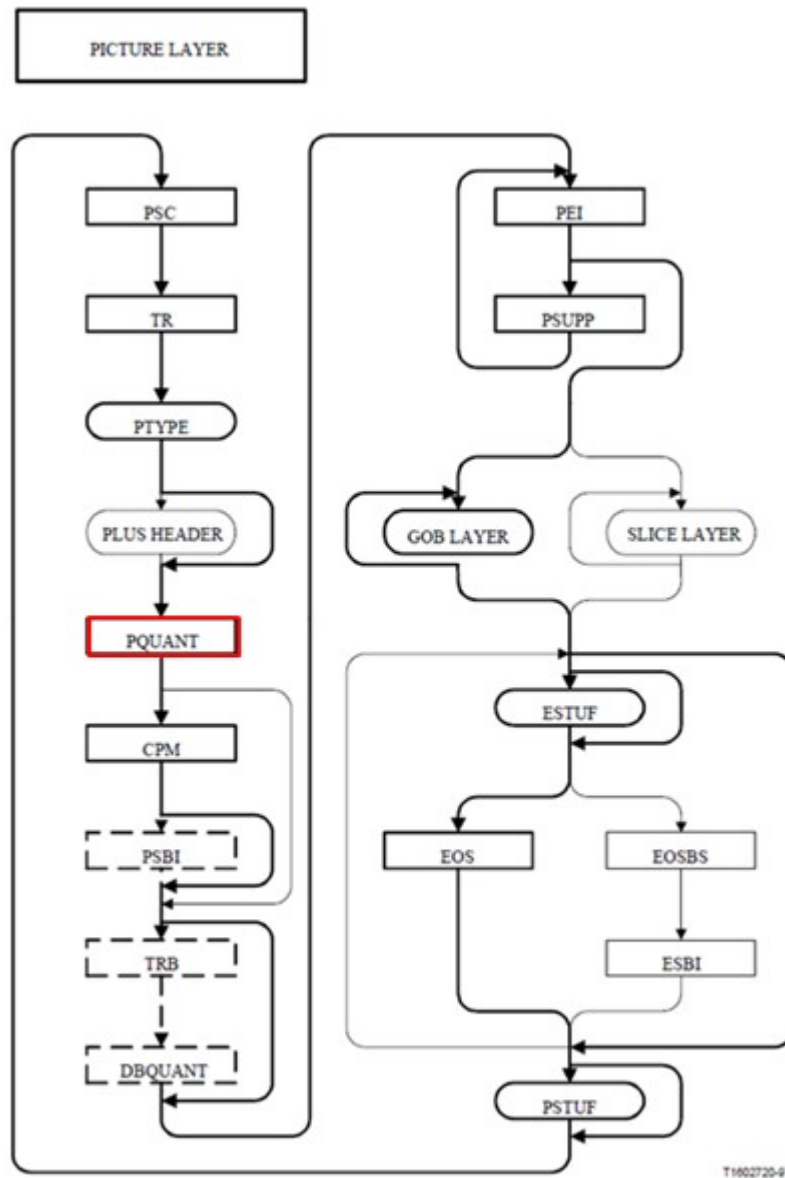


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

c. Claim 3

H.263 discloses or suggests claim 3’s limitation. Saber, ¶292.

H.263 describes that “[t]he video coder provides a self-contained digital bitstream which may be combined with other multi-facility signals (for example as defined in Recommendation H.223). The video decoder performs the reverse process.” H.263, 2. H.263’s Figure 1 (annotated below) illustrates that the video “Source coder” (red) passes data to a “Transmission buffer” (blue) that is output as a “Coded bitstream” (purple), and that a “Coded bitstream” is received by a “Receiving buffer” (green) before it is passed to a “Source decoder” (light blue).

Thus the “Coded bitstream” (“*encoded bit-stream*”) is “*transmitted*” to a decoder. Saber, ¶¶293-94.

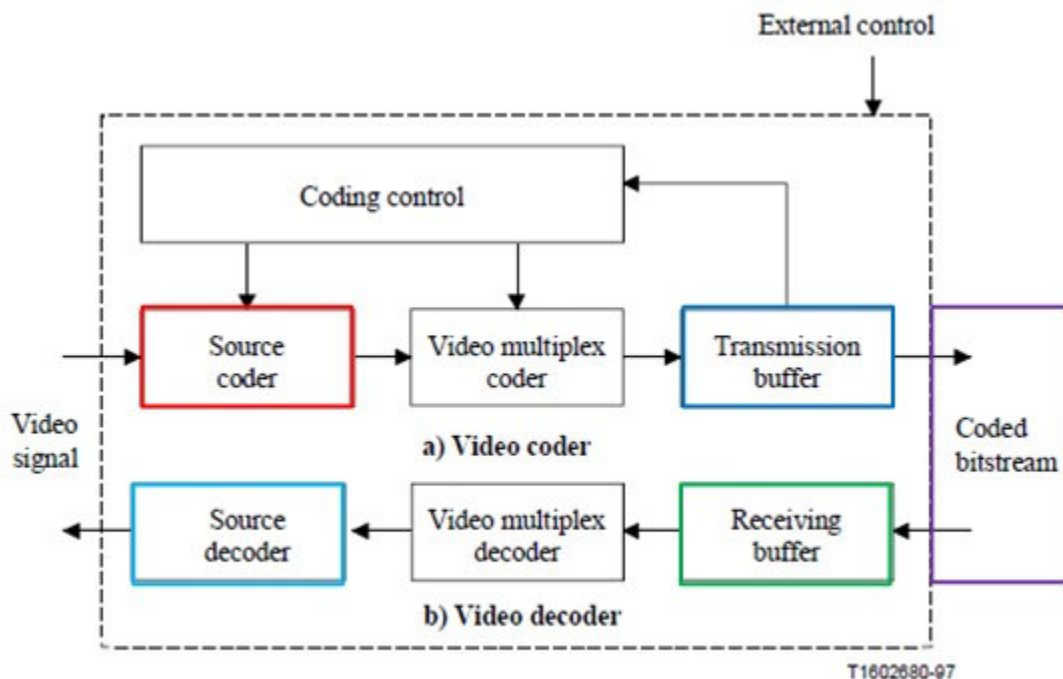


Figure 1/H.263 – Outline block diagram of the video codec

H.263 thus discloses or suggests that the bitstream is “*transmitted from a video encoding device to a corresponding video decoding device.*” Saber, ¶295.

d. Claim 4

H.263 discloses or suggests claim 4’s limitation. Saber, ¶296.

As described in Sections VII.B.2.a(iv) and VII.B.2.a(v), H.263 discloses that PQUANT (“*default level of quantization*”) is part of a picture header. And “[d]ata for each picture consists of a picture header followed by data for [GOBs] or for slices.” H.263, 22. In other words, PQUANT is provided with each picture. Saber, ¶297.

H.263 operates on pictures occurring at a Picture Clock Frequency (PCF) (approximately 29.97 times per second). H.263, 7. H.263’s Figure 6, annotated below, illustrates that each picture of a video sequence occurring at a PCF (green) contains picture layer data including PQUANT (red). Saber, ¶298. PQUANT thus specifies the “*default level of quantization*” for each picture. Saber, ¶298. Further, a POSITA would have understood that PQUANT of a first picture could specify a first value for QUANT used for the first picture, and PQUANT of a second picture could specify a second (and different) value for QUANT used for the second picture. Saber, ¶299. H.263 therefore discloses/suggests the “*default level of quantization is updated during the encoding of the digital video sequence*” (e.g.,

using PQUANT with each picture), and “providing an indication of the updated default level of quantization to a decoding process.” Saber, ¶¶299-300.

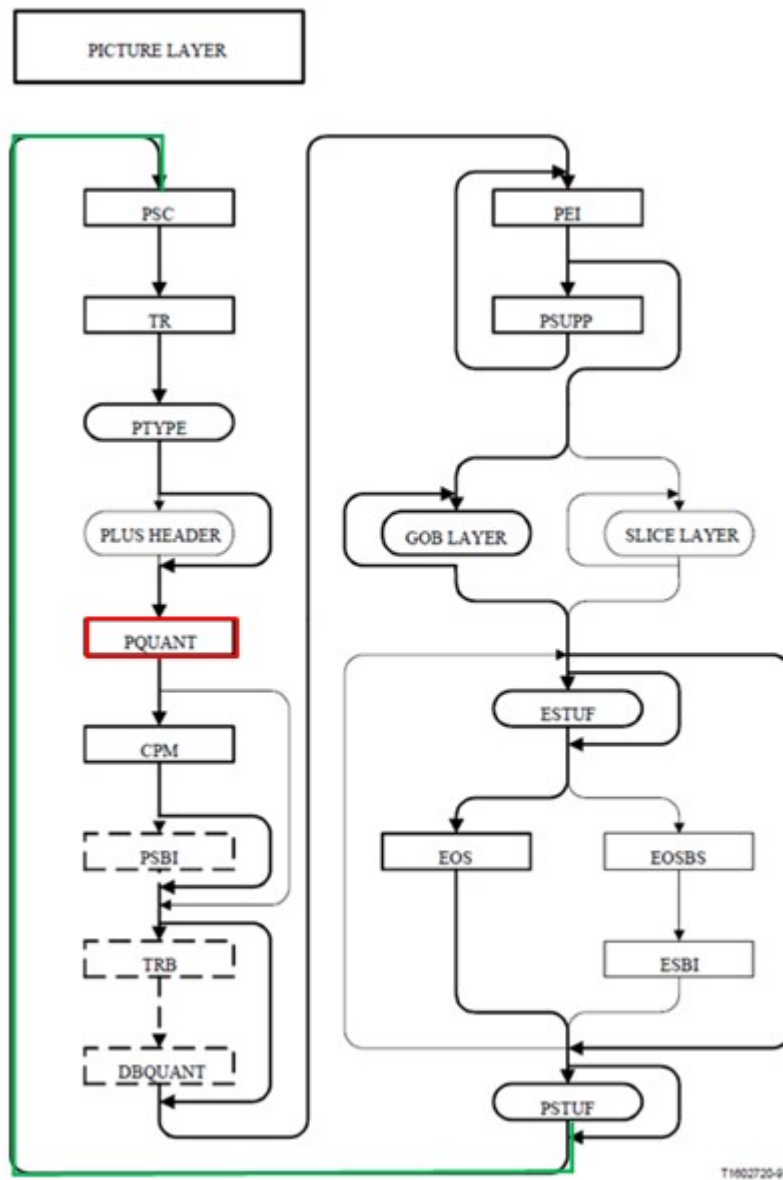


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

e. Claim 5

H.263 discloses or suggests claim 5’s limitation. Saber, ¶301.

As described in §VII.B.2.a(iv), H.263's PQUANT defines "*a default level of quantization.*" H.263 describes that PQUANT "indicates the quantizer QUANT to be used for the picture until updated by any subsequent GQUANT or DQUANT." H.263, 30. GQUANT (specified at GOB layer) and/or DQUANT (specified at macroblock layer) thus can change the value of QUANT—used for quantizing the blocks within a picture—from the value previously specified for that picture (by PQUANT). H.263, 32-33, 37; Saber, ¶302. And as explained below, both GQUANT and DQUANT disclose or suggest quantizing "*based on a level of quantization different from the default level of quantization*" (e.g., PQUANT). Saber, ¶302.

GQUANT is part of the GOB header and sets QUANT in place of the value previously set by PQUANT. H.263, 32-33; Saber, ¶¶303-04. GQUANT therefore causes quantizing "*based on a level of quantization different from the default level of quantization*" (e.g., PQUANT). Saber, ¶304.

DQUANT discloses or suggests the above-described limitation of claim 5. Saber, ¶305. As reflected in H.263's Table 9 (annotated below), DQUANT is included in certain types of macroblocks, including the INTER macroblock type named "INTER+Q." Saber, ¶305.

Table 9/H.263 – Macroblock types and included data elements for normal pictures

Picture type	MB type	Name	COD	MCBPC	CBPY	DQUANT	MVD	MVD ₂₋₄
INTER	Not coded	–	X					
INTER	0	INTER	X	X	X		X	
INTER	1	INTER+Q	X	X	X	X	X	
INTER	2	INTER4V	X	X	X		X	X
INTER	3	INTRA	X	X	X			
INTER	4	INTRA+Q	X	X	X	X		
INTER	5	INTER4V+Q	X	X	X	X	X	X
INTER	Stuffing	–	X	X				
INTRA	3	INTRA		X	X			
INTRA	4	INTRA+Q		X	X	X		
INTRA	Stuffing	–		X				

NOTE – “X” means that the item is present in the macroblock.

When DQUANT is present for a macroblock, it is included in the macroblock’s header as a “a two-bit code [used] to define a change in QUANT.” H.263, 33-34, 37, Fig. 10. DQUANT therefore discloses/suggests quantizing “*based on a level of quantization different from the default level of quantization*” (e.g., PQUANT). Saber, ¶306.

H.263 discloses or suggests the limitations of claim 5. Saber, ¶307. As described in §VII.B.2.a(iii), H.263 discloses producing “*sets of transform coefficient values*” that are “*representative of said blocks of prediction error values.*” And H.263 describes that “[t]ransmitted blocks are transformed and resulting coefficients are quantized.” H.263, 11. In other words, H.263 quantizes the transform coefficient values (representing prediction error values) to produce quantized coefficient values. Saber, ¶308. And as described in §VII.B.2.a(iv),

H.263 quantizes using QUANT, where GQUANT “indicates the quantizer QUANT,” and DQUANT “define[s] a change in QUANT.” H.263, 33, 37. H.263 therefore discloses/suggests that “*sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values*” using a quantizer based on GQUANT or DQUANT. Saber, ¶308.

f. Claim 6

H.263 discloses or suggests claim 6’s limitation. Saber, ¶309.

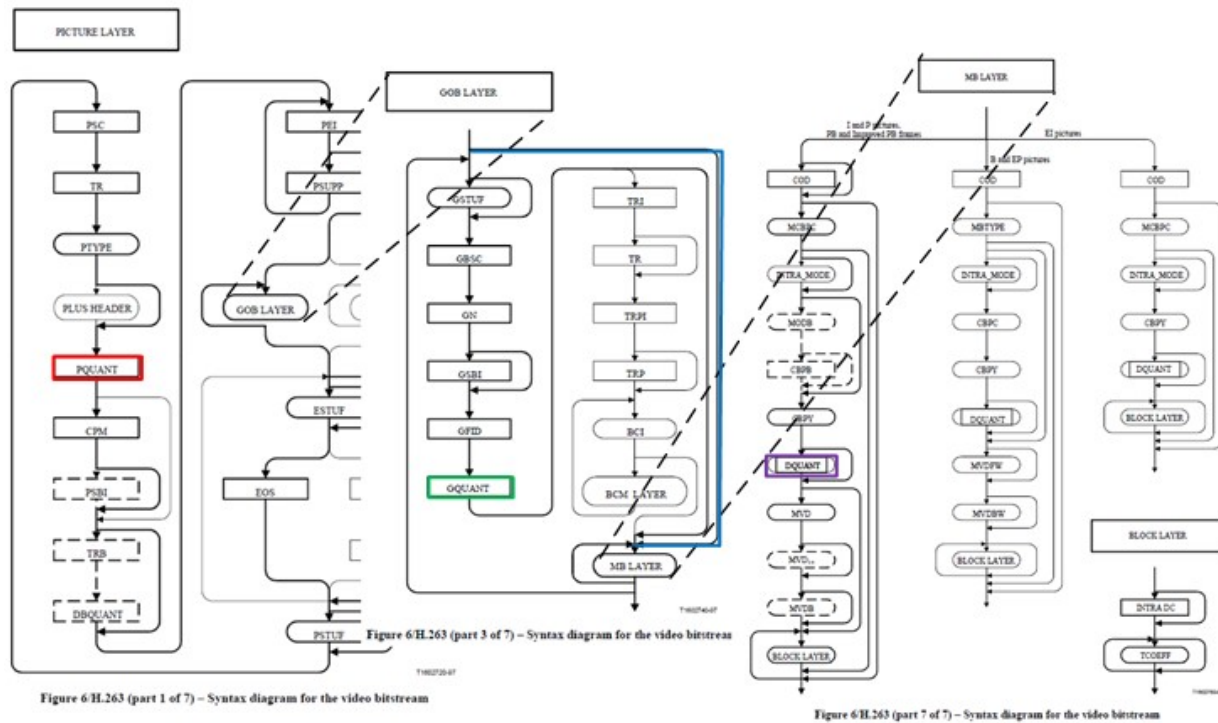
As described in §VII.B.2.e, for example, H.263’s DQUANT defines a change to a previously specified (e.g., by PQUANT) QUANT (“*default level of quantization*”). DQUANT is therefore a differential value (“*an indication of a difference*”). Saber, ¶¶310-11. The QUANT changed by DQUANT is “*said level of quantization.*” Saber, ¶310. “DQUANT is a two-bit code to define a change in QUANT. In Table 12 [reproduced below] the differential values for the different codewords are given.” H.263, 37.

Table 12/H.263 – DQUANT codes and differential values for QUANT

Index	Differential value	DQUANT
0	−1	00
1	−2	01
2	1	10
3	2	11

H.263 describes how “QUANT ranges from 1 to 31; if the value for QUANT after adding the differential value is less than 1 or greater than 31, it is clipped to 1 and 31 respectively.” H.263, 37. That is, DQUANT changes QUANT by specifying a value (selected from -2, -1, 1, and 2) that is added to QUANT. Saber, ¶312. DQUANT thus specifies the difference between a prior value of QUANT (set, for example, previously by PQUANT) and the new value of QUANT. Saber, ¶312.

Further, H.263 GOB headers may be omitted from transmission or may be empty. H.263, 32. Consequently, there are situations in which no GQUANT is coded after transmission of PQUANT (specifying QUANT for a picture). Saber, ¶313. Figure 6 of H.263 (annotated below) illustrates how, at the GOB layer, no GQUANT (green) may be included in the video bitstream (blue path), such that when DQUANT is coded at the macroblock layer (purple), it changes the value of QUANT set by PQUANT for that picture (red) (“*the default level of quantization*”). Saber, ¶314. H.263’s DQUANT therefore discloses/suggests “*an indication of a difference between said level of quantization and the default level of quantization.*” Saber, ¶314.



DQUANT is part of a macroblock header and included in the video bitstream. H.263, 21, 33, Fig. 6 (part 7 of 7), Fig. 10. H.263 therefore discloses/suggests “providing” DQUANT “to a decoding process.” Saber, ¶315.

g. Claim 7

H.263 discloses or suggests claim 7’s limitation. Saber, ¶316.

As described in §VII.B.2.e, for example, H.263’s DQUANT (specified at the macroblock layer) defines a change to a previously specified (e.g., by PQUANT) QUANT (“default level of quantization”). The QUANT changed by DQUANT is “an actual level of quantization applied to the sets of transform coefficients that is different from the default level of quantization.” Saber, ¶317. H.263 describes that its video syntax is “arranged in a hierarchical structure,” with the picture layer at

the top, the GOB layer below the picture layer, and the macroblock layer below the GOB layer. H.263, 13. As illustrated in H.263's Figure 6, annotated below, the picture layer of a video bitstream includes PQUANT for a picture (red) followed by the GOB layer data for all of the GOBs forming the picture (blue). Saber, ¶318.

Further, each GOB optionally includes a header (as described in §VII.B.2.a(iv) and §VII.B.2.f) as well as the macroblock layer data for all macroblocks of the GOB (purple). Saber, ¶318. The syntax repeats (green) for the next picture in the video. Saber, ¶318.

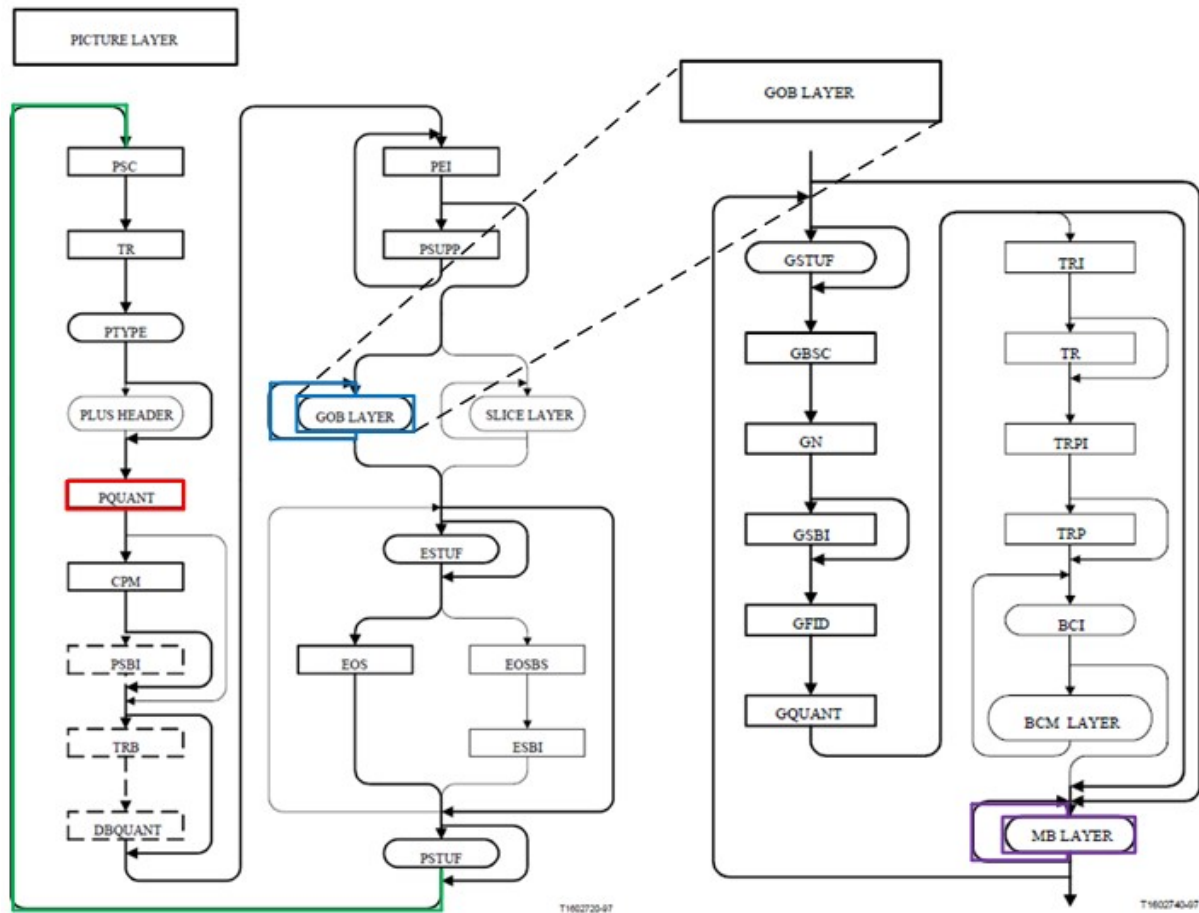


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

Figure 6/H.263 (part 3 of 7) – Syntax diagram for the video bitstream

Thus, every picture is transmitted with PQUANT, which sets QUANT for the picture (“*default level of quantization*”). Saber, ¶319. When the first macroblock of the first GOB of the picture—which does not include a GOB header (H.263, 32)—includes DQUANT to specify a change to the picture’s QUANT (“*an actual level of quantization that is different from the default level of quantization*”), the changed QUANT can be applied to the entire picture (“*applied to the sets of transform coefficients for a particular frame*”) when no other QUANT changes are specified for the picture (e.g., no subsequent use of GQUANT or DQUANT in the

picture). H.263, 37; Saber, ¶319. In this manner, a first DQUANT provided in the bitstream and associated with a first picture (e.g., in the header of the first macroblock of the first picture) sets a first “*level of quantization*” for that picture, and a second DQUANT provided in the bitstream and associated with the first macroblock of a second picture sets a second “*level of quantization*” for that picture. Thus, DQUANT may specify quantization adjustments “*from one frame...to another.*” Saber, ¶320. The PQUANT of the first and second pictures may have the same value such that both specify the same “*default level of quantization*” subsequently adjusted by DQUANT. Saber, ¶320.

H.263 therefore discloses/suggests a level of quantization that is “adjusted from one frame of the digital video sequence to another such that an actual level of quantization applied to the sets of transform coefficients for a particular frame of the digital video sequence is different from the default level of quantization” by the DQUANT parameter. Saber, ¶321.

h. Claim 8

H.263 discloses or suggests claim 8’s limitation. Saber, ¶322. As described in Sections VII.B.2.g and VII.B.2.f, H.263’s DQUANT updates QUANT (an “*actual level of quantization*”) by specifying a differential value added to QUANT (e.g., as set by PQUANT) (“*representable as a difference with respect to the default level of quantization*”). Saber, ¶323.

i. Claim 9

H.263 discloses or suggests claim 9's limitation. Saber, ¶324.

As described in Sections VII.B.2.g and VII.B.2.f, H.263's DQUANT *“provid[es] to a decoding process an indication of a difference between said actual level of quantization and the default level of quantization.”* Saber, ¶325.

j. Claim 10

H.263 discloses or suggests the limitations of claim 10 via both DQUANT and GQUANT. Saber, ¶326.

As described in §VII.B.2.e, H.263's GQUANT (specified at the GOB layer) and DQUANT (specified at the macroblock layer) both update QUANT from its previous value (e.g., as specified by PQUANT) (*“the default level of quantization”*). As described in §VII.B.2.e, the updated QUANT (whether specified by GQUANT or DQUANT) discloses an *“actual level of quantization applied to the sets of transform coefficients”* that is *“different from the default level of quantization.”* And as described in §VII.B.2.a(i), H.263 discloses luminance and chrominance blocks (*“blocks of pixels”*) that combine to form macroblocks, further combined to form GOBs, of a picture (*“into which a frame of said digital video sequence is divided”*). Saber, ¶327.

H.263 describes that *“a macroblock consists of four luminance blocks and the two spatially corresponding colour difference blocks,”* and *“each macroblock*

relates to 16 pixels by 16 lines of Y and the spatially corresponding 8 pixels by 8 lines of C_B and C_R.” H.263, 11. Each H.263 macroblock thus corresponds to a “*segment*” into which a “*plurality of blocks of pixels*” (e.g., four luminance blocks and two chrominance blocks) “*are grouped*.” Saber, ¶¶328-30. Further, as described in §VII.B.2.f, DQUANT is part of a macroblock header, so each macroblock (the recited “*segment*”) that includes DQUANT to update QUANT (“*actual level of quantization*”) enables different QUANT adjustments with different macroblocks (“*level of quantization*” that is “*adjusted from one segment of a frame to another*”). Saber, ¶331.

H.263’s GQUANT discloses or suggests the above-described limitations of claim 10. Saber, ¶332. H.263 describes that “[e]ach GOB is divided into macroblocks,” and that a GOB comprises one to four macroblock rows, depending on the source picture format. H.263, 7, 11; Saber, ¶332. Each H.263 GOB therefore discloses/suggests a “*segment*” into which a “*plurality of blocks of pixels*” (e.g., a plurality of macroblocks, each of which groups four luminance blocks and two chrominance blocks) “*are grouped*.” Saber, ¶332. Further, as described in §VII.B.2.e, GQUANT is part of a GOB header, so each GOB (the recited “*segment*”) that includes GQUANT (“*level of quantization*”) enables different values of QUANT for different GOBs (“*level of quantization*” that is “*adjusted from one segment of a frame to another*”). Saber, ¶332.

As described in §VII.B.2.a(iv) and §VII.B.2.e, the updated QUANT (whether updated by DQUANT or GQUANT) (“*actual level of quantization*” that is “*for a particular segment of a frame*”) is used to quantize transform coefficient values (“*applied to the sets of transform coefficients*”). Saber, ¶333.

k. Claim 11

H.263 discloses or suggests claim 11’s limitation. Saber, ¶¶334-35. As described in §VII.B.2.j and §VII.B.2.f, DQUANT discloses “*providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization.*”

3. Claim 12 and Dependents

l. Claim 12

i. 12[pre]

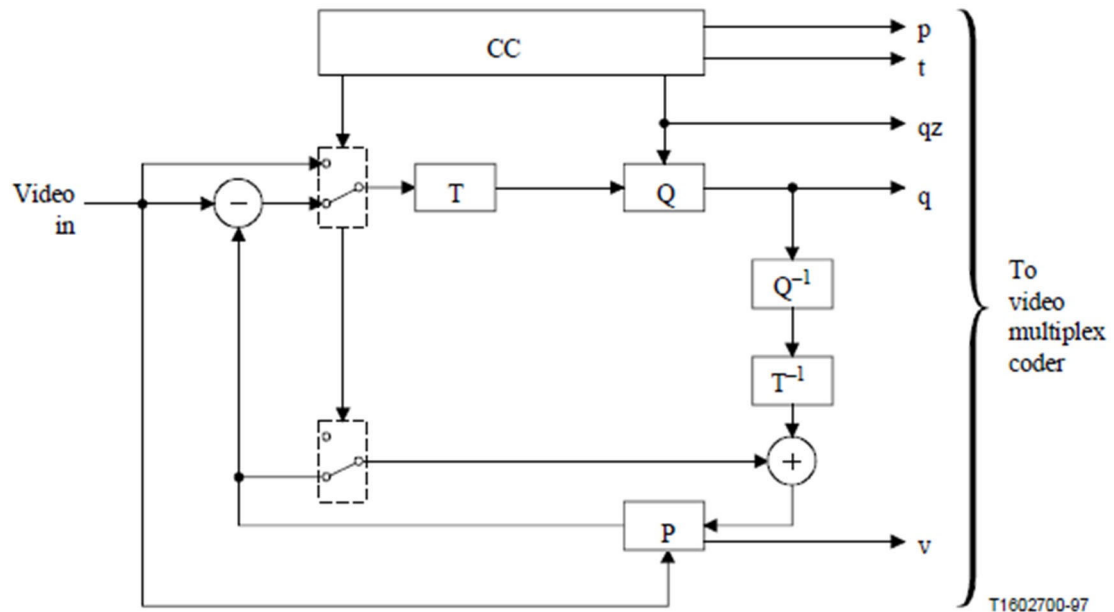
If limiting, H.263 discloses or suggests 12[pre]. Saber, ¶336.

As described in §VII.B.2.a(i), H.263 discloses a method “*to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels.*” Saber, ¶337.

As illustrated in Figure 3 (reproduced below), H.263’s “Source coder” receives video input (“Video in”) and produces outputs that pass to a “video

multiplex coder.” Saber, ¶338. The elements of the source coder include

“prediction, block transformation and quantization.” H.263, 9.



- T Transform
- Q Quantizer
- P Picture Memory with motion compensated variable delay
- CC Coding control
- p Flag for INTRA/INTER
- t Flag for transmitted or not
- qz Quantizer indication
- q Quantizing index for transform coefficients
- v Motion vector

Figure 3/H.263 – Source coder

Figure 1 of H.263 (reproduced below) illustrates that the Source coder is part of a video codec that receives a “Video signal” and, in combination with the video multiplex coder and a transmission buffer, produces a “Coded bitstream.” Saber, ¶339.

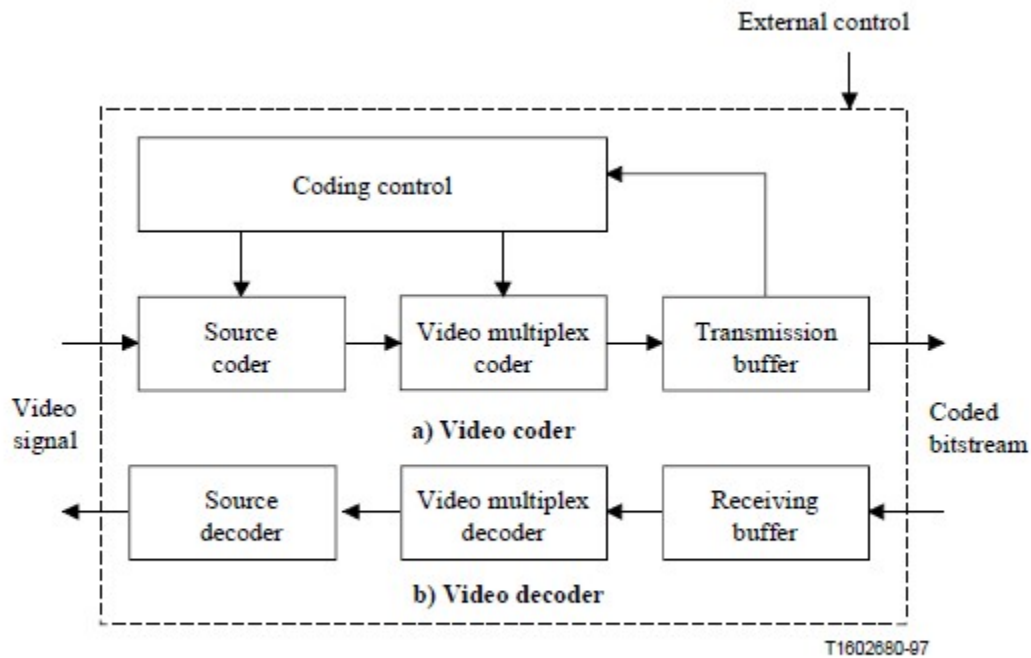


Figure 1/H.263 – Outline block diagram of the video codec

The H.263 source coder is therefore a “*video encoder for encoding a digital video sequence.*” Saber, ¶340.

ii. 12[a]

H.263 discloses or suggests 12[a]. Saber, ¶341.

As described in §VI.A.1, the function of this means-plus-function limitation is “*applying motion compensated prediction to blocks of pixels and provid[ing] corresponding blocks of prediction error values,*” and the corresponding structure is a motion compensated predictor and combiner. ’148Pat., 7:66-8:4 (motion compensated predictor), 8:5-9 (combiner); Saber, ¶342.

As described in §VII.B.2.a(ii), H.263 discloses “*applying motion compensated prediction to blocks of pixels.*” H.263’s Figure 3 (annotated below) illustrates that the source coder includes a “Picture Memory with motion compensated variable delay” (green) that generates motion vectors (light blue) based on video input (orange). Saber, ¶343. The “Picture Memory with motion compensated variable delay” generates additional data (purple), which a POSITA would have recognized is a prediction formed by performing motion compensation using the generated motion vectors and the Picture Memory. Saber, ¶¶343-47 (in H.261, on which H.263 is based, the illustrated coder includes an identically-named “Picture memory with motion compensated variable delay”—also illustrated as “Frame Memory,” “Motion Estimation,” and “Motion Compensation”—that generates a “Predictive frame”); *see also* H.263, 11 (“The primary form of prediction is inter-picture and may be augmented by motion compensation”). A POSITA would thus have understood that the “Picture Memory with motion compensated variable delay” is a motion compensated predictor that applies motion-compensated prediction to blocks of pixels in the same manner as the motion compensated predictor in the ’148 Patent. Saber, ¶347.

H.263 discloses “*provid[ing] corresponding blocks of prediction error values*” for the reasons described in §VII.B.2.a(ii). Saber, ¶348. H.263’s Figure 3 (below) illustrates that the source coder takes the difference (red) between the

video input (orange) and the result (purple) of motion-compensated prediction (green). Saber, ¶348. As described in §VII.B.2.a(ii), this difference (blue) produces “*blocks of prediction error values.*” Saber, ¶348. Further, the ’148 Patent explains that a combiner subtracts predicted pixel values from input pixel values. ’148Pat., 8:5-9. And similarly in H.263, the difference block (red) subtracts predicted pixel values (the result of motion-compensated prediction) from input pixels (i.e., input pictures), thus operating in the same manner as the combiner of the ’148 Patent. Saber, ¶¶349-50. A POSITA therefore would have known that the “*blocks of prediction error values*” are produced in the H.263 source coder by a combiner (red) in the same manner as the combiner described by the ’148 Patent. Saber, ¶351.

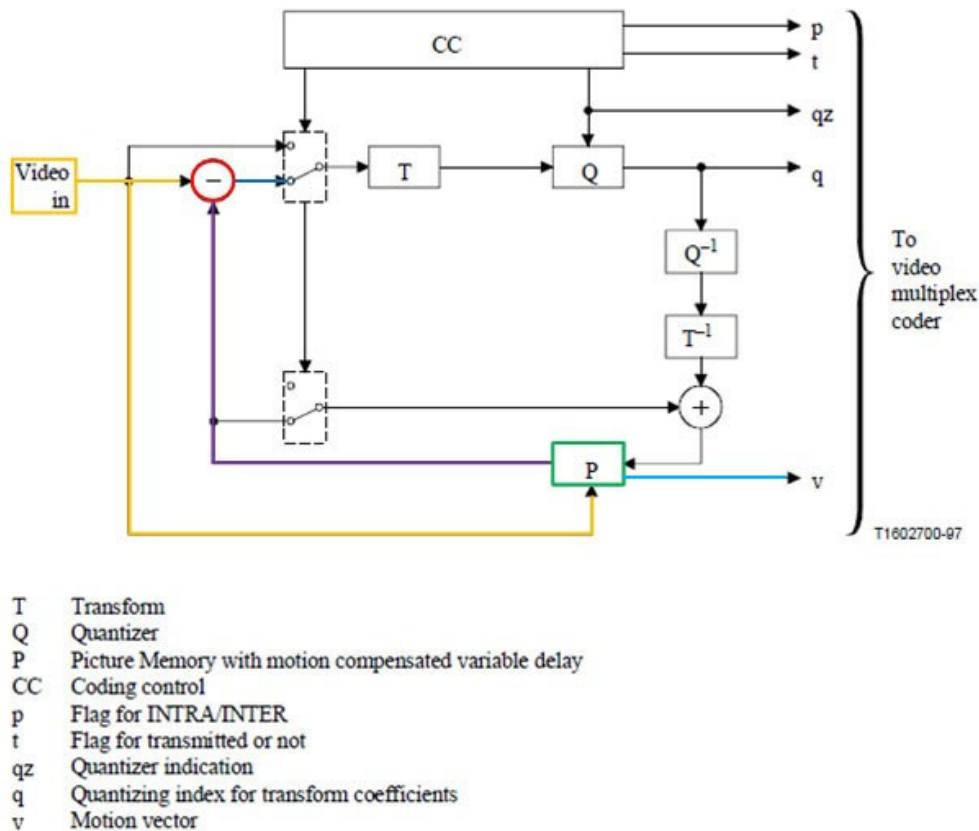


Figure 3/H.263 – Source coder

iii. 12[b]

H.263 discloses or suggests 12[b]. Saber, ¶352.

As described in §VI.A.2, the function of this means-plus-function limitation “providing sets of transform coefficient values representative of said blocks of prediction error values,” and the structure is the transform block. ’148Pat., 18:6619:3; Saber, ¶353.

H.263 discloses/suggests “transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said

blocks of prediction error values” for the reasons described in §VII.B.2.a(iii).

Saber, ¶354. H.263’s Figure 3 (annotated below) illustrates that the source coder

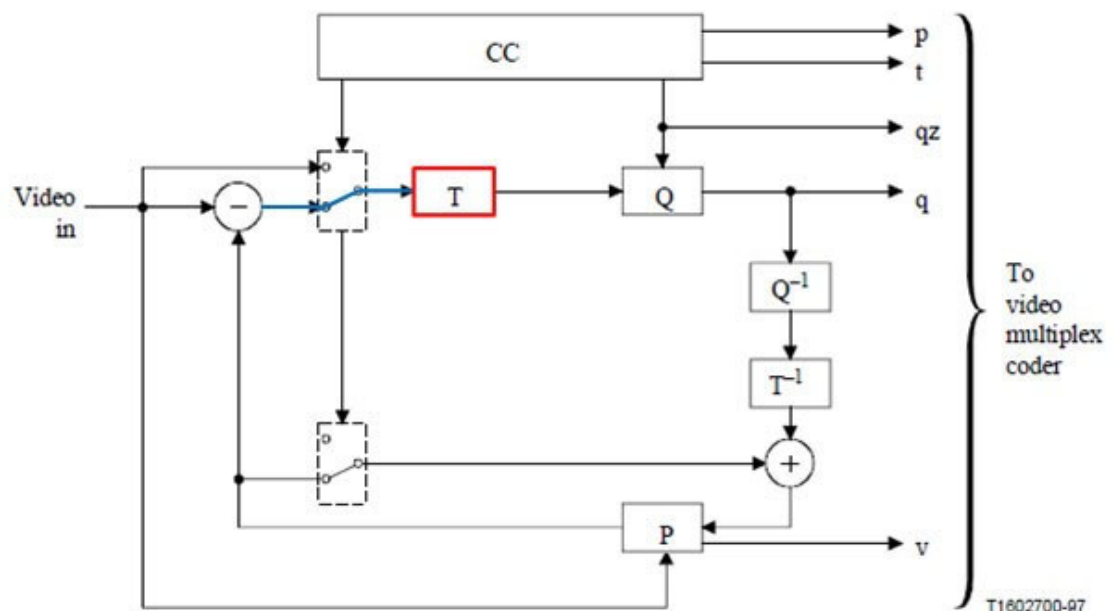
(“*video encoder*”) includes a “Transform” block (red) that receives blocks of

prediction error values (blue). Saber, ¶355. A POSITA would have understood that

the Transform block “*provid[es] sets of transform coefficient values*” from the

received prediction error values in the same manner as the transform block in the

’148 Patent. Saber, ¶355.



T	Transform
Q	Quantizer
P	Picture Memory with motion compensated variable delay
CC	Coding control
p	Flag for INTRA/INTER
t	Flag for transmitted or not
qz	Quantizer indication
q	Quantizing index for transform coefficients
v	Motion vector

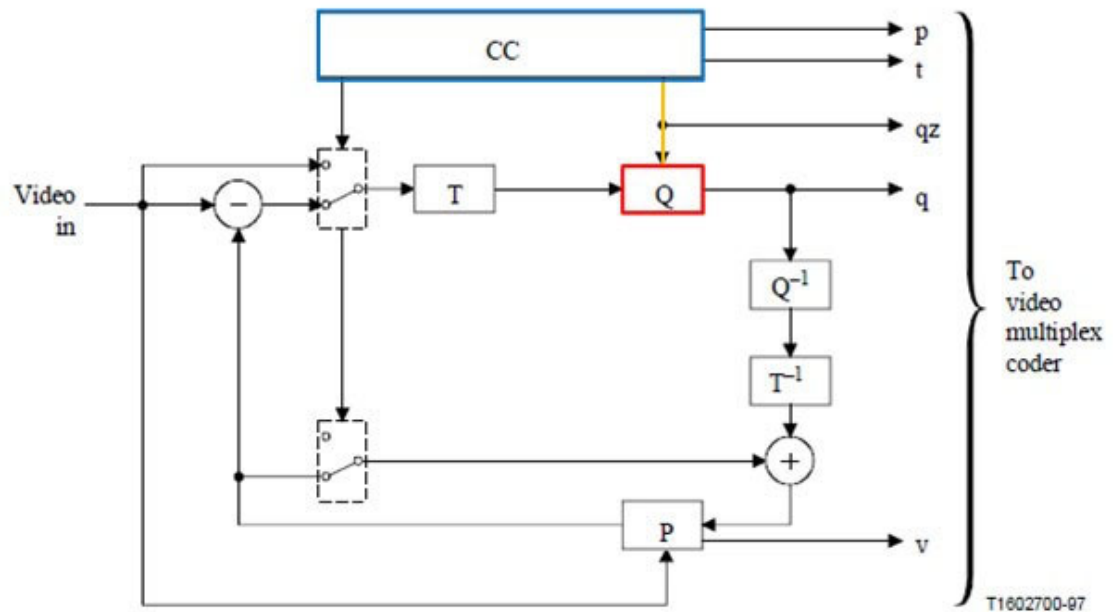
Figure 3/H.263 – Source coder

iv. 12[c]

H.263 discloses or suggests 12[c]. Saber, ¶356.

As described in §VI.A.3, the function of this means-plus-function limitation is “*selecting a default level of quantization for quantizing said sets of transform coefficient values*,” and the structure is the control manager.’148Pat., 6:16-22; 19:60-65; Saber, ¶357.

H.263 discloses/suggests “*selecting a default level of quantization for quantizing said sets of transform coefficient values*” for the reasons described in §VII.B.2.a(iv). Saber, ¶358. H.263 describes, under the heading of “Coding control,” that “[s]everal parameters may be varied to control the rate of generation of coded video data. These include...the quantizer.” H.263, 13. Further, H.263’s Figure 3 (annotated below) illustrates that the source coder (“*video encoder*”) includes a “Coding control” block (blue) that provides a “Quantizer indication” (orange) to a “Quantizer” (red). Saber, ¶¶359-61. Since H.263 describes that “Coding control” included varying parameters such as those for quantization and illustrates that the “Coding control” block provides the “Quantizer indication,” it would have been obvious to a POSITA to use the “Coding control” block to *select* the default level of quantization. Saber, ¶362. The “Coding control” block of H.263 operates in the same manner as the control manager in the ’148 Patent. Saber, ¶362.



T Transform
Q Quantizer
P Picture Memory with motion compensated variable delay
CC Coding control
p Flag for INTRA/INTER
t Flag for transmitted or not
qz Quantizer indication
q Quantizing index for transform coefficients
v Motion vector

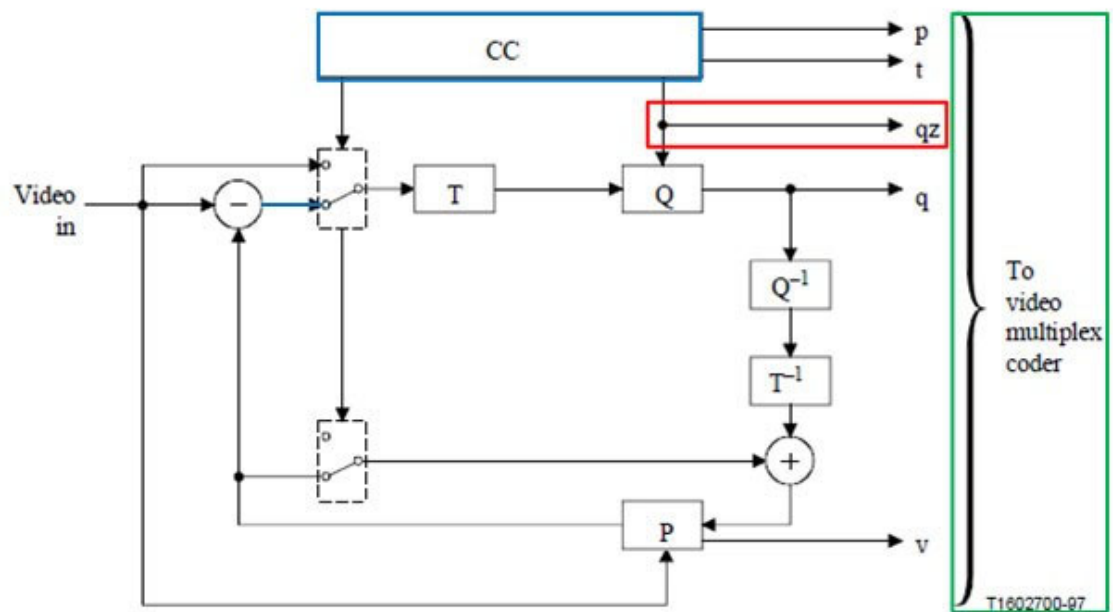
Figure 3/H.263 – Source coder

v. 12[d]

H.263 discloses or suggests 12[d]. Saber, ¶363.

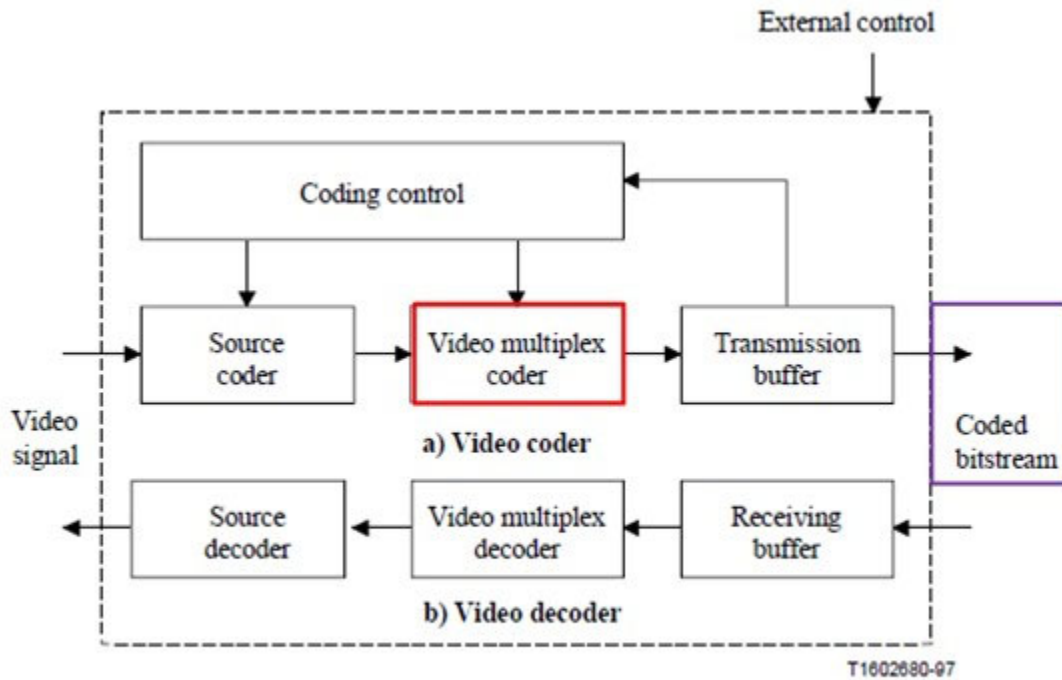
As described in §VI.A.4, the function of this means-plus-function limitation is “*providing an indication of said default level of quantization in the encoded bitstream,*” and the structure is the video multiplex coder. ’148Pat., 6:38-47; 19:60-65; Saber, ¶364.

H.263 discloses/suggests “*providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons described in §VII.B.2.a(v), and for the substantially similar limitation discussed therein. Saber, ¶365. H.263’s Figure 3 (below and annotated) illustrates that a quantizer indication (red), from the coding control block (blue), is provided to a video multiplex coder (green). Saber, ¶366. And H.263’s Figure 1 (below and annotated) illustrates that the video multiplex coder (red) produces a coded bitstream (purple). Saber, ¶367. A POSITA would have understood that H.263’s video multiplex coder operates in the same manner as the video multiplex coder in the ’148 Patent to provide an encoded bitstream “*representative of the digital video sequence*” that includes “*an indication of said default level of quantization.*” Saber, ¶368.



- T Transform
- Q Quantizer
- P Picture Memory with motion compensated variable delay
- CC Coding control
- p Flag for INTRA/INTER
- t Flag for transmitted or not
- qz Quantizer indication**
- q Quantizing index for transform coefficients
- v Motion vector

Figure 3/H.263 – Source coder



m. Claim 13

H.263 discloses or suggests claim 13's limitation. Saber, ¶369.

H.263 discloses/suggests “*wherein said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from the default level of quantization*” for the reasons described in §VII.B.2.e.

H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263’s video source coding, so the source coder performs the disclosed encoding method. H.263, 9; Saber, ¶¶370-71.

n. Claim 14

H.263 discloses or suggests claim 14's limitation. Saber, ¶372.

H.263 discloses/suggests “*arranged to provide an indication of a difference between said level of quantization and the default level of quantization in the encoded bit-stream*” for the reasons described in §VII.B.2.f. Saber, ¶373.

H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263's video source coding, and therefore the source coder conforms to H.263's encoding method that provides the indication “*in the encoded bit-stream*” sent to a decoder. H.263, 9; Saber, ¶374.

o. Claim 15

H.263 discloses or suggests claim 15's limitation. Saber, ¶375.

H.263 discloses/suggests “*wherein the default level of quantization can be updated and the video encoder is arranged to provide an indication of the updated default level of quantization in the encoded bit-stream*” for the reasons described in §VII.B.2.d. Saber, ¶376. H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263's video source coding, so the source coder conforms to H.263's encoding method that provides the indication “*in the encoded bit-stream*” sent to a decoder. H.263, 9; Saber, ¶377.

p. Claim 16

H.263 discloses or suggests claim 16's limitation. Saber, ¶378.

As discussed in §VII.B.3.b, H.263 discloses a video encoder in which “*said sets of transform coefficient values are quantized to yield sets of quantized transform coefficient values representative of said blocks of prediction error values based on a level of quantization different from*” (e.g., based on GQUANT or DQUANT) a previously set QUANT. As discussed in §VII.B.3.d, H.263 discloses an “*updated default level of quantization.*” A POSITA would therefore have understood that GQUANT or DQUANT provide “*a level of quantization different*” from PQUANT values of different frames (“*the updated default level of quantization*”). Saber, ¶¶379-80.

q. Claim 17

H.263 discloses or suggests claim 17’s limitation. Saber, ¶381.

H.263 “specifies a coded representation” for “compressing the moving picture component of audio-visual services at low bit rates.” H.263, i. For example, H.263 could be used with different television standards. H.263, 2. A POSITA would have understood that “audio-visual services” (audio and visual data) represents multimedia data. H.263 is one of a series of recommendations (“Series H”) directed to “[a]udiovisual and multimedia systems.” H.263, 155. It therefore would have been obvious to a POSITA to provide the H.263 source coder (“*video encoder*”) in a “*multimedia terminal*” to encode the video component of

audiovisual and multimedia data (e.g., a terminal producing television data). Saber, ¶¶382-84.

r. Claim 18

H.263 discloses or suggests claim 18's limitation. Saber, ¶385.

As described in §VII.B.3.f, H.263 describes it is used for the coded representation of the video component of television data. A POSITA would have known that an encoded H.263 data stream could be transmitted wirelessly over a radio channel (e.g., to a television or personal computer) via "*a radio telecommunications device*." Saber, ¶386.

H.263 references Recommendation H.324 ("Terminal for low bit rate multimedia communication") "[f]or a complete overview of possible picture formats and video coding algorithms" that may be supported by H.263 encoders and decoders. H.263, 8-9. H.324 illustrates that the H.263 video codec may be provided in a multimedia system, such as a "visual telephone system," that operates on "mobile radio networks." H.324, 1, Fig. 1. A POSITA would have understood that a visual telephone system operating on a mobile radio network is a "*radio telecommunications device*." Saber, ¶387.

s. Claim 19

H.263 discloses or suggests claim 19's limitation. Saber, ¶388.

H.263 discloses/suggests “*wherein said level of quantization applied to said sets of transform coefficient values can be adjusted from one frame of the digital video sequence to another so as to apply an actual level of quantization to the sets of transform coefficients for a particular frame of the digital video sequence and wherein said actual level of quantization is different from the default level of quantization*” for the reasons described in §VII.B.2.g. Saber, ¶389. H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263’s video source coding, so the source coder conforms to H.263’s encoding method. H.263, 9; Saber, ¶390.

t. Claim 20

H.263 discloses or suggests claim 20’s limitation. Saber, ¶391.

H.263 discloses/suggests “*provid[ing] an indication of said difference with respect to the default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons described in §VII.B.2.i. Saber, ¶392.

H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263’s video source coding, so the source coder conforms to H.263’s encoding method. H.263, 9; Saber, ¶393.

u. Claim 21

H.263 discloses or suggests claim 21’s limitation. Saber, ¶394.

The table below demonstrates each of the elements of claim 21 is substantially similar to an element of claims 10 or 11 (both of which are Challenged Claims, *see* §VII.B.2.j and §VII.B.2.k, respectively). Saber, ¶¶395-96.

As some examples, claim 21 recites “arranged to group the plurality of blocks...divided into one or more segments,” and claim 10 recites an encoding method in which “the plurality of blocks of pixels...are grouped into one or more segments.” Additionally, claim 21 recites “adjust[ing] the level of quantization...from one segment of a frame to another,” and claim 10 recites an encoding method in which “the level of quantization...is adjusted from one segment of a frame to another.” H.263 therefore discloses/suggests claim 21’s limitation for the reasons discussed in §VII.B.2.j and §VII.B.2.k. H.263 explains that the source coder (“video encoder”) shows in “generalized form” the components of H.263’s video source coding, so the source encoder (“video encoder”) performs H.263’s encoding method. H.263, 9; Saber, ¶397.

Claim 21	Corresponding Limitation from Claim 10/11
21[a]	wherein the plurality of blocks of pixels, into which a frame of said digital video sequence is divided, are grouped into one or more segments; (§VII.B.2.j)
21[b]	wherein the level of quantization applied to said sets of transform coefficient values is adjusted from one segment of a frame to another; (§VII.B.2.j)

Claim 21	Corresponding Limitation from Claim 10/11
21[c]	such that an actual level of quantization applied to the sets of transform coefficients for a particular segment of a frame is different from the default level of quantization; (§VII.B.2.j)
21[d]	providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization. (§VII.B.2.k)

v. Claim 22

H.263 discloses or suggests claim 22's limitation. Saber, ¶398.

H.263 discloses/suggests “*provid[ing] an indication of said difference with respect to the default level of quantization in the encoded bit-stream representative of the digital video sequence*” for the reasons described in §VII.B.2.k. Saber, ¶399.

H.263 explains that the source coder (“*video encoder*”) shows in “generalized form” the components of H.263's video source coding, so the source coder conforms to H.263's encoding method. H.263, 9; Saber, ¶400.

VIII. CONCLUSION

Inter partes review of the Challenged Claims is respectfully requested.

IX. MANDATORY NOTICES

A. Real Party in Interest

Petitioner identifies itself as real parties in interest.

B. Related Matters

To the best of Petitioner's knowledge, the '148 patent has been involved in the following district court litigation:

- *Element Television Company, LLC et al v. Nokia Corporation a/k/a Nokia of America Corporation et al.*, Case No. 0:24-cv-04269 (D.M.N. Nov. 25, 2024);
- *Amazon.com, Inc. et al v. Nokia Technologies Oy*, IPR2024-01176 (P.T.A.B. Aug. 30, 2024); and
- *Nokia Technologies Oy v. Amazon.com, Inc.*, No. 1-23-cv-01236 (D. Del. Oct. 31, 2023).

C. Notice of Counsel and Service Information

Lead Counsel	Back-Up Counsel
Brian E. Ferguson Reg. No. 36,801 WINSTON & STRAWN LLP 1901 L Street, N.W. Washington, D.C. 20036 T: 202-282-5276 beferguson@winston.com	Eimeric Reig (<i>pro hac vice</i> to be submitted) WINSTON & STRAWN LLP 101 California Street 21 st Floor San Francisco, CA 94111 T: 415-591-6808 ereigplessis@winston.com
Back-Up Counsel	Back-Up Counsel
Joseph C. Masullo (<i>pro hac vice</i> to be submitted) WINSTON & STRAWN LLP 1901 L Street, N.W. Washington, D.C. 20036 T: 202-282-5897 jmasullo@winston.com	Lauren M. Rennecker (<i>pro hac vice</i> to be submitted) WINSTON & STRAWN LLP 1901 L Street, N.W. Washington, D.C. 20036 T: 202-282-5897 lrennecker@winston.com

Dated: June 16, 2025

Respectfully Submitted,

/s/ Brian E. Ferguson

Brian E. Ferguson
Reg. No. 36,801
WINSTON & STRAWN LLC
1901 L Street, N.W.
Washington, D.C. 20036
T: 202-282-5276
beferguson@winston.com

CERTIFICATE OF COMPLIANCE

I hereby certify that this brief complies with the type-volume limitations of 37 C.F.R. § 42.24, because it contains 13,962 words (as determined by the Microsoft Word word-processing system used to prepare the brief and including annotations to the figures), excluding the parts of the brief exempted by 37 C.F.R. § 42.24.

Dated: June 16, 2025

Respectfully Submitted,

/s/ Brian E. Ferguson

Brian E. Ferguson

Reg. No. 36,801

WINSTON & STRAWN LLC

1901 L Street, N.W.

Washington, D.C. 20036

T: 202-282-5276

beferguson@winston.com

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6(e), I hereby certify that on this 16th day of June, 2025, I caused to be served a true and correct copy of the foregoing and any accompanying exhibits by priority mail on the following:

Nokia Corporation and Alston & Bird LLP
Vantage South End
1120 South Tryon Street
Suite 300
Charlotte, NC 28203-6818

Dated: June 16, 2025

Respectfully Submitted,

/s/ Brian E. Ferguson
Brian E. Ferguson
Reg. No. 36,801
WINSTON & STRAWN LLC
1901 L Street, N.W.
Washington, D.C. 20036
T: 202-282-5276
beferguson@winston.com