

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

**DECLARATION OF DR. ELI SABER, PH.D.
REGARDING U.S. PATENT NO. 8,175,148**

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	BACKGROUND AND QUALIFICATIONS	2
III.	LEVEL OF ORDINARY SKILL IN THE ART	7
IV.	MATERIALS CONSIDERED AND RELIED UPON.....	8
V.	LEGAL STANDARDS.....	9
A.	Legal Standards for Priority Date	10
B.	Legal Standards for Obviousness.....	10
VI.	SUMMARY OF OPINIONS	14
VII.	TECHNOLOGY OVERVIEW	14
VIII.	OVERVIEW OF THE '148 PATENT.....	17
A.	Subject Matter Overview of the '148 Patent	17
B.	Brief Summary of '148 Patent Prosecution History	20
IX.	CLAIM CONSTRUCTION	22
A.	Means-Plus-Function (MPF)	22
1.	<i>“Means for Encoding” / “Module for Encoding”</i>	22
2.	<i>“Means for Transforming” / “Module for Transforming”</i>	24
3.	<i>“Means for Selecting” / “Module for Selecting”</i>	26
4.	<i>“Means for Providing”</i>	28
X.	DETAILED EXPLANATION OF GROUNDS	29

A.	Ground 1: Obviousness over MPEG-1	29
1.	MPEG-1	29
2.	Claim 1 and Dependents.....	38
3.	Claim 12 and Dependents.....	68
4.	Claim 23.....	92
B.	Ground 2: Obviousness over H.263	95
1.	H.263.....	95
2.	Claim 1 and Dependents.....	99
3.	Claim 12 and Dependents.....	135
XI.	CONCLUSION	164

LISTING OF CHALLENGED CLAIMS

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 bit-stream representative of the digital video sequence.
Claim 3	
3	An encoding method according to claim 2, wherein the encoded bit-stream 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, Eli Saber, hereby declare as follows:

I. INTRODUCTION

1. I have been retained on behalf of Snap Inc. (“Snap” or “Petitioner”) to offer technical opinions related to U.S. Patent No. 8,175,148 (“the ’148 patent”) (EX-1001). I understand that Snap is requesting that the Patent Trial and Appeal Board (“PTAB” or “Board”) institute an inter partes review (“IPR”) proceeding for the ’148 patent. Specifically, I have been asked to consider the validity of claims 1-23 of the ’148 patent (the “Challenged Claims”) in view of prior art and obviousness considerations from the perspective of a person of ordinary skill in the art (“POSITA”) at the time of the invention as it relates to the ’148 patent.

2. I am being compensated for my work in this matter at my standard consulting rate of \$625 per hour for such consulting services. I am also being reimbursed for expenses that I incur during the course of this work. My compensation is not contingent upon the results of my study, the substance of my opinions, or the outcome of any proceeding involving the Challenged Claims. I have no financial interest in the outcome of this matter or in the pending litigation between Petitioner and Patent Owner.

3. My analysis here is based on my years of education, research and experience, as well as my investigation and study of relevant materials, including those cited herein.

4. I may rely upon these materials, my knowledge and experience, and/or

additional materials to rebut arguments raised by the Patent Owner. Further, I may also consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

5. My analysis of the materials in this proceeding is ongoing, and I will continue to review any new material as it is provided. This declaration represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my opinions stated herein based on new information and on my continuing analysis of the materials already provided.

II. BACKGROUND AND QUALIFICATIONS

6. I am over the age of 18 and am competent to write this declaration. I have personal knowledge, or have developed knowledge of these technologies, based upon education, training, or experience, of the matters set forth herein.

7. My qualifications are set forth in my curriculum vitae, a copy of which is filed as Exhibit EX-1004. The following is a brief summary of my relevant qualifications and professional experience.

8. I am a Professor in the Department of Electrical and Microelectronic Engineering (EME) and the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology (RIT) in Rochester, New York. I also serve as the Director of the Image, Video and Computer Vision Laboratory. I joined RIT's full-time faculty in 2004. I have over thirty-four (34) years of industry and academic experience, thirty (30) of which are in the fields of Computer Vision, Digital

Image/Video Processing, Pattern Recognition, and Machine Learning.

9. I received a Ph.D. in Electrical Engineering from the University of Rochester in 1996. My concentration was in Signal/Image/Video Processing, Pattern Recognition, and Computer Vision. I received a Master of Science Degree in Electrical Engineering from the University of Rochester in 1992 in the same concentration as listed above, and I received a Bachelor of Science Degree in Electrical and Computer Engineering, Summa Cum Laude, from the State University of New York at Buffalo in 1988.

10. Before becoming a full-time professor, I worked for Xerox Corporation for 16 years, from 1988 to 2004. During my years at Xerox, I was responsible for delivering color management, image processing innovations, architectures, and algorithms; xerographic sub-systems for a variety of color products; and control systems for toner production facilities. One of several roles I held at Xerox was Advanced Development Scientist and Manager. In that capacity, I established the Advanced Design Laboratory—an Imaging/Xerographics lab—and provided technical and managerial leadership for the Electrical, Imaging and Xerographics Department. In another role, as Product Development Scientist and Manager, I led the research and development of image quality metrics for various product platforms. I also led the Image Science, Analysis, and Evaluation area, with 12 to 15 direct reports and a budget of approximately \$2 million.

11. From 1997 until 2004, I was an adjunct faculty member at the

Electrical Engineering Department of the Rochester Institute of Technology and at the Electrical & Computer Engineering Department of the University of Rochester. I was responsible for teaching undergraduate and graduate coursework such as Digital Signal Processing, Digital Image Processing, Digital Video Processing, Pattern Recognition, Detection and Estimation Theory and Communication Systems; and performing research in multimedia applications, pattern recognition, image understanding and color engineering.

12. Since joining RIT full time, I have been responsible for teaching several undergraduate and graduate courses in Digital Signal Processing, Digital Image Processing, Digital Video Processing, Advanced Neural Networks and Deep Learning, Pattern Recognition, Engineering Analysis, Random Signal & Noise, Advanced Engineering Mathematics, Matrix Methods, Communications, Modern Control Theory, and Linear Systems.

13. I proposed, developed, and introduced the Digital Video Processing and the Advanced Neural Network and Deep Learning courses; and I reestablished the Pattern Recognition course at the Electrical Engineering Department at RIT. These graduate courses are instrumental in providing MS/PhD students with the proper foundation and knowledge in all aspects of the subject areas in order to perform in-depth research in the field and continue to advance the state of the art.

14. The Digital Video Processing course is designed to provide a thorough and comprehensive coverage of the field. Topics covered include the mathematical

representation of digital video including motion and time-varying image formation models, spatio-temporal sampling/reconstruction, sampling structure conversions, motion estimation using optical flow, block-based, pel-recursive, Bayesian and point correspondence type methodologies, segmentation, motion tracking, filtering, restoration, super-resolution and image/video compression.

15. The Pattern Recognition course, on the other hand, covers pertinent concepts in Bayes decision theory, linear and nonlinear classifiers, artificial neural networks, feature selection/generation/extraction and various types of clustering algorithms and techniques.

16. The “Advanced Neural Networks and Deep Learning: Theory, Mathematics and Algorithms” course is intended to provide an in-depth comprehensive study of the theory, mathematics and algorithms for advanced neural networks and their usage in deep learning. It is primarily targeted for upper graduate students that are planning to pursue advanced studies/research in the field. The course begins with an in-depth study of the mathematical methodologies for neural networks and optimization (zero-, first- and second-order) techniques. It then progresses to provide a thorough study of shallow and deep neural networks architectures and algorithms, activation functions, regression, two- and multi-class classification, unsupervised learning, feature engineering/selection, backpropagation along with applications to a variety of engineering problems.

17. I am currently working to develop a follow-on course (part 2) to the

Advanced Neural Networks and Deep Learning course described above. This course is intended to cover convolutional and recurrent style neural networks, principles of feature engineering, kernel methods, tree-based learners, attention mechanisms/transformers, reinforcement learning and generative adversarial networks. This course is being targeted for the first time in Fall 2026.

18. I am also currently working to revamp the graduate course titled: “Image and Video Compression”. The updated version of this course will start by providing a brief history of video coding and video coding standards and then progress to discussing video basics, formats and fundamentals of video compression. It will then progress to cover JPEG, JPEG 2000, H.261, MPEG-1, MPEG-2, H.263, MPEG-4 visual, H.264 and H.265 and if time permits, AV-1 and H.266. The course is intended to debut in Fall 2027.

19. My current research focuses on machine learning for object tracking/remote sensing applications; image/video processing for multimedia, military & biomedical applications; and computer vision and three-dimensional scene reconstruction. As a principal investigator (PI) or co-principal investigator, I have acquired almost \$6 million in research funding since joining RIT and have managed multiple government grants from the Department of Defense as well as several corporate grants from Hewlett-Packard, Lenel, and Data Physics. I am currently managing (as PI) a multiyear government grant from the Department of Defense titled “Target Detection/Tracking and Activity Recognition from

Multimodal Data.” In this project, we propose to develop operational target detection/tracking techniques and activity recognition/understanding methods that can leverage multimodal data in a fusion framework using deep learning algorithms while providing accurate and near real-time performance.

20. In 2012, I was awarded the prestigious Trustees Scholarship Award, the highest award at RIT with regard to research recognition.

21. I am the author or co-author of thirty-eight (38) peer-reviewed journal publications.

22. I have also authored or co-authored 102 conference and workshop publications and a book entitled Advanced Linear Algebra for Engineers with MATLAB, published by CRC Press in February 2009, and am a named inventor on multiple U.S. and foreign patents.

23. I am a senior member of the Institute of Electrical and Electronic Engineers and a member of the IEEE Signal Processing Society, the Electrical Engineering Honor Society, Eta Kappa Nu.

III. LEVEL OF ORDINARY SKILL IN THE ART

24. In rendering the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of a POSITA at the time of the alleged invention, which I understand to be April 23, 2002. I understand that the factors considered in determining the ordinary level of skill in a field of art include the level of education and experience of persons working in the field, the

types of problems encountered in the field, the teachings of the prior art, and the sophistication of the technology at the time of the alleged invention. I understand that a POSITA is not a specific real individual, but rather is a hypothetical individual having the qualities reflected by the factors above. I understand that a POSITA would also have knowledge from the teachings of the prior art, including the art cited below.

25. Taking these factors into consideration, in my opinion, a POSITA at the time of the '148 Patent would have had a (1) Bachelor's degree in Electrical Engineering, Computer Engineering, Computer Science, or a comparable field of study such as Physics, and (2) approximately 2-3 years of practical experience with video and/or image processing or coding. More education could substitute for experience, and vice versa.

26. Before April 23, 2002, my level of skill in the art was at least that of a POSITA. I am qualified to provide opinions concerning what a POSITA would have known and understood at that time, and my analysis and conclusions herein are from the perspective of a POSITA as of that date.

IV. MATERIALS CONSIDERED AND RELIED UPON

27. In forming my opinions expressed herein, in addition to relying on my education and experience, I considered the totality of the following materials:

Exhibit	Reference
1001	U.S. Patent No. 8,175,148 ("the '148 patent" or "'148Pat.")

Exhibit	Reference
1002	File History of the '148 patent (“’148FH”)
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, <i>Image and Video Compression Standards, Algorithms and Architectures</i> , (Kluwer Academic Publishers, 1995) (“Bhaskaran”)
1012	ITU-T Recommendation H.261 (March 1993) (“H.261”)

I have also considered any materials cited in this declaration even if not included in the list above.

V. LEGAL STANDARDS

28. I am not a lawyer and do not provide any legal opinions, but I have been advised that certain legal standards are to be applied by technical experts in forming opinions regarding the meaning and validity of patent claims. I have applied the legal standards described below.

29. It is my understanding that assessing the validity of a U.S. patent based on a prior art analysis involves two steps. First, one must assess what meaning a POSITA would have given the claim terms. Second, after assessing the meaning of

the claim terms, one may then assess validity by comparing a patent claim to the prior art. I understand that the teaching of the prior art is viewed through the eyes of a POSITA at the time the invention was made. My analysis as to what constitutes a relevant POSITA is set forth above.

A. Legal Standards for Priority Date

28. I understand that the “priority date” (or “earliest effective filing date”) of a patent is the date on which it is filed, or the date on which an earlier application was filed if the patentee properly claims the benefit of the earlier application’s filing date. For purposes of my analysis in this declaration, I have been asked to assume that the ’148 patent is entitled to a priority date of April 23, 2002.

B. Legal Standards for Obviousness

29. My understanding is that a patent claim is invalid as obvious only if the subject matter of the claimed invention “as a whole” would have been obvious to a POSITA at the time the invention was made. To determine the difference(s) between a prior art reference (or a proposed combination of prior art references) and the claims, the question of obviousness is not whether the differences themselves would have been obvious, but whether the claimed invention as a whole would have been obvious. Also, obviousness grounds cannot be sustained by mere conclusory statements. Rather, it is necessary to provide some articulated reasoning with rational underpinning to support the legal conclusion of obviousness.

30. I understand that a patent claim that comprises several elements is not

proved obvious by simply showing that each of its elements was independently known in the prior art. In my evaluation of whether any claim of the '148 patent would have been obvious, I considered whether a POSITA would have had a reason with a rational basis that would have motivated a POSITA to combine the elements or concepts from the prior art in the same way as in the claimed invention.

31. It is my understanding that there is no single way to define the line between true inventiveness on one hand—which is patentable—and the application of common sense and ordinary skill to solve a problem on the other hand—which is not patentable. For instance, factors such as market forces or other design incentives may be the source of what produced a change, rather than true inventiveness.

32. I understand that assessing obviousness may consider whether the change was merely the predictable result of using prior art elements according to their known functions, or whether it was the result of true inventiveness. And the obviousness assessment may also consider whether there is some teaching or suggestion in the prior art to make the modification or combination of elements recited in the claim at issue. Also, the decision maker may consider whether the innovation applies a known technique that had been used to improve a similar device or method in a similar way. The decision maker may also consider whether the claimed invention would have been obvious to try, meaning that the claimed innovation was one of a relatively small number of possible approaches to the problem with a reasonable expectation of success by those skilled in the art.

33. I understand that in considering obviousness, it is important to be careful not to engage in hindsight and that many true inventions might seem obvious after the fact. I understand that obviousness is considered from the perspective of a POSITA at the time the claimed invention was made, and not based on what is known today or what is learned from the teaching of the patent.

34. I understand that the following factors must be evaluated to determine whether a claim is obvious: the scope and content of the prior art; the difference or differences, if any, between the claim of the patent and the prior art; the level of ordinary skill in the art at the time the claimed invention was made; and, if available, the objective indicia of non-obviousness, also known as “secondary considerations.”

35. I understand that the secondary considerations can include: commercial success of a product due to the merits of the claimed invention; a long felt need for the solution provided by the claimed invention; unsuccessful attempts by others to find the solution provided by the claimed invention; copying of the claimed invention by others; unexpected and superior results from the claimed invention; acceptance by others of the claimed invention as shown by praise from others in the field or from the licensing of the claimed invention; teaching away from the conventional wisdom in the art at the time of the invention; independent invention of the claimed invention by others before or at about the same time as the named inventor thought of it; and other evidence tending to show obviousness.

36. I understand that, to establish a secondary consideration, evidence must demonstrate a nexus between that secondary consideration and the claimed invention.

37. I understand that a petition must identify, for any terms with means-plus-function limitations, the specific portions of the specification that describe the structure, material, or acts corresponding to each claimed function. I also understand a rebuttable presumption exists that with use of the term “means,” a claim term is interpreted as means-plus-function under 35 U.S.C. § 112.

38. I also understand that a petitioner seeking to demonstrate that the prior art discloses a means-plus-function limitation must prove that the corresponding structure—or an equivalent—was present in the prior art and performing the same claimed function.

39. I understand that the standard for determining whether the prior art discloses the corresponding structure or an equivalent is whether there are insubstantial differences, which means that if the assertedly equivalent structure performs the claimed function in substantially the same way to achieve substantially the same result as the corresponding structure described in the specification, it is equivalent.

40. I understand that proving structural equivalence does not require a component-by-component analysis, but rather that it is sufficient to show structure, materials, or acts that perform the claimed function in substantially the same way to

achieve substantially the same result because the claim limitation is the overall structure corresponding to the claimed function and not individual components.

VI. SUMMARY OF OPINIONS

41. For the reasons I discuss below, it is my opinion that claims 1-23 of the '148 patent would have been obvious to a POSITA in view of the prior art.

VII. TECHNOLOGY OVERVIEW

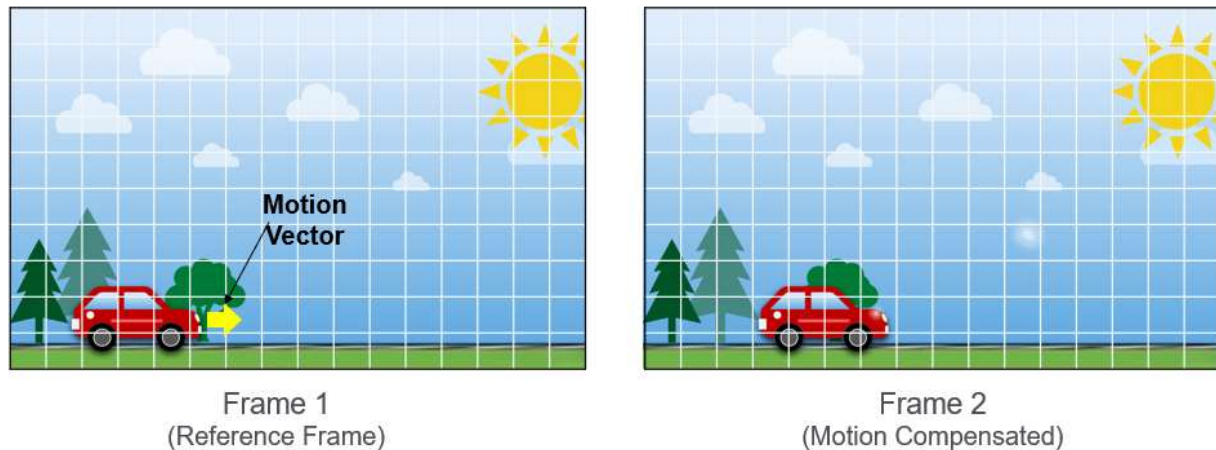
42. A video sequence is formed of images, or frames, displayed one after the other at a rate (e.g., between 15 and 30 frames per second) to create the illusion of motion. '148Pat., 1:20-24. Because of the large amount of data required to transmit video data in uncompressed form (e.g., as uncompressed frames), various video compression techniques are used to reduce redundancies in a video sequence. '148Pat., 2:25-34; 2:45-52.

43. One such technique is “‘block-based transform coding’, in which a mathematical transformation, such as a two-dimensional Discrete Cosine Transform (DCT), is applied to blocks of image pixels. This transforms the image data from a representation comprising pixel values to a form comprising a set of coefficient values representative of spatial frequency components,” and reduces 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.

44. Another technique is “‘motion-compensated prediction’, [which is

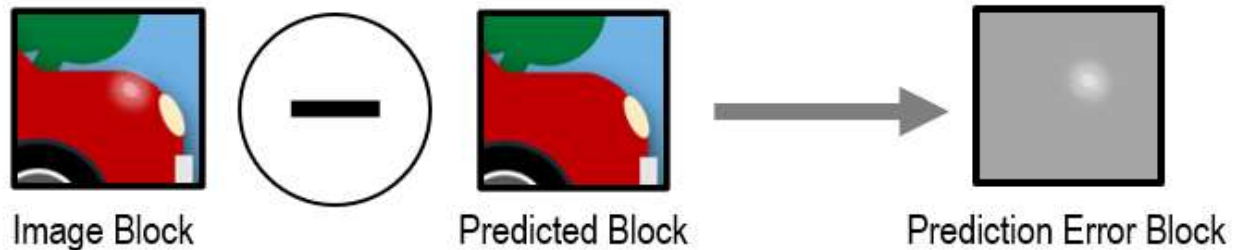
used] to reduce the temporal redundancy in [a video] sequence” by predicting a frame using “motion vectors” that characterize the motion of objects in other, previously coded, “reference” frames. ’148Pat., 3:12-22. Motion-compensated prediction may be performed on individual portions of a frame, known as “macroblocks.” ’148Pat., 1:51-56, 3:24-26. “Frames of a video sequence, compressed using motion-compensated prediction, are generally referred to as INTER-coded[.]” ’148Pat.,. 3:27-29.

45. The figure below illustrates motion-compensated prediction, where a macroblock in a current frame (“Frame 2”) is predicted using a motion vector that characterizes the motion of a macroblock in a reference frame (“Frame 1”).



46. According to the Background section of ’148Pat, “[m]otion-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 INTER-coded frame. The prediction error 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.

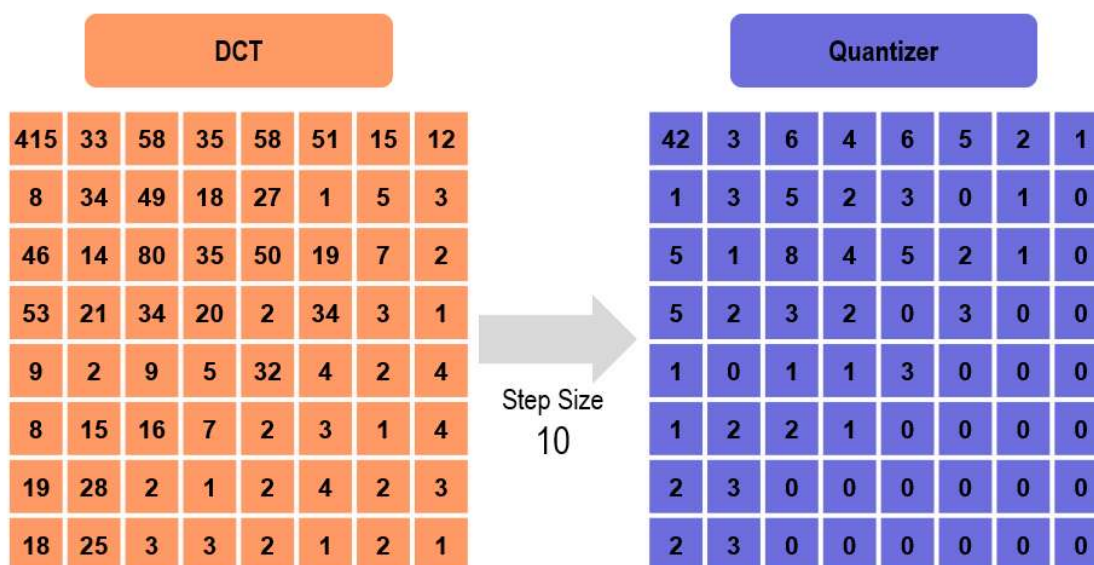


47. The prediction error frame has characteristics similar to a still image, and therefore block-based transform coding such as a DCT can be applied to it (producing DCT coefficients for each prediction error block) to reduce its spatial redundancy. ’148Pat., 3:40-44, 6:10-15.

48. To reduce the number of bits required to represent an image, the Background explains that the DCT coefficients (produced from image data for an INTRA-coded frame, or from prediction error for an INTER-coded frame) are quantized to produce quantized DCT coefficients. ’148Pat., 4:21-31, 4:47-50, 6:16-20. “[Q]uantization of the DCT coefficients is performed by dividing each coefficient value by the quantization parameter QP and rounding the result to the nearest integer. In this way the quantization process yields a set of quantized DCT coefficient values that have a reduced numerical precision compared with the coefficient values originally generated by DCT transformation block...[T]he quantized DCT coefficients may be represented by a smaller number of data bits than required to represent the corresponding coefficient before quantization.”

'148Pat., 4:33-42. The Background additionally explains how quantization can be used to adjust the number of bits required to encode a video. Increasing QP provides a more compressed video at the expense of quality, while decreasing QP provides higher quality video that is less compressed. '148Pat., 4:52-62, 6:22-27.

49. The figure below illustrates the result of quantization, in which DCT coefficients are quantized using a step size of 10 (i.e., each coefficient is divided by 10 and rounded to the nearest integer) to produce quantized coefficients.



VIII. OVERVIEW OF THE '148 PATENT

A. Subject Matter Overview of the '148 Patent

50. The '148 Patent is titled "Method and Device for Indicating Quantizer Parameters in a Video Coding System" and issued on May 8, 2012, from Application 11/881,367 ("367 application"), filed on July 26, 2007. '148Pat., (21), (22), (45), (54). The '148 Patent claims priority to Provisional Application No.

60/374,667, filed April 23, 2002, and is a division of U.S. Patent Application No. 60/374,667 (“’629 application”), filed April 23, 2003. ’148Pat., (60), (62).

51. A significant portion of the ’148 Patent’s specification is devoted to the “Background of the Invention” section, which provides an overview of video coding and discusses purported shortcomings of prior art solutions. ’148Pat., 1:20-11:21. For example, the Background section describes the structure and function of prior art generic video encoders and decoders, including the manner in which inter-coded and intra-coded frames of a video are encoded and decoded. ’148Pat., 3:45-10:43. The Background section additionally provides an overview of certain prior art video coding solutions developed by various standards bodies and/or industry organizations. ’148Pat., 10:44-11:17.

52. More specifically, the Background of the ’148 Patent describes quantizing the DCT coefficients by a quantization parameter QP to reduce the number of bits required to represent an image. ’148Pat., 4:20-32, 4:47-50, 6:16-20. ’The 148 Patent describes that a decoder needs to know the quantization parameter QP value that was used when an encoder performed quantization, and that typically the QP value is transmitted to the decoder once per “slice” (where a slice contains part of a frame, and is formed from macroblocks in consecutive coding order). ’148Pat., 2:5-11, 10:33-40.

53. The Background further explains that some prior art solutions, such as the H.26L video coding recommendation presented in Joint Model Number 1

(“JM1”) from the Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, transmit the QP value for each picture, and optionally for each slice forming the picture.

’148Pat., 10:44-60. In H.26L, the QP value may also be adjusted at the macroblock layer using a quantizer change parameter (DQUANT). ’148Pat., 11:8-14.

54. The ’148 Patent purports to improve upon the prior art by introducing a sequence level quantization parameter (SQP) transmitted by an encoder in a bitstream (e.g., in the sequence header). ’148Pat., 11:25-27, 18:9-16, 19:60-67.

55. The ’148 Patent provides the following overview:

According to the invention, when starting to encode a new video sequence, encoder 600 determines a default or reference level of quantization that is to be used throughout encoding of the video sequence to quantize the DCT coefficient values generated in quantizer 106. Throughout the following description, this default or reference level of quantization will be termed a “sequence level quantization parameter”, or SQP for short. The choice of an SQP for a given video sequence is controlled by control manager 660 and, for example, may be based on consideration of the properties of the sequence to be encoded and the bandwidth available for transmission of the encoded bit-stream produced by the encoder.

’148Pat., 19:4-16.

56. The SQP thereby describes a default level of quantization that is to be used throughout encoding of the video sequence. ’148Pat., 19:4-11. Further, the actual QP to be used for quantization can be adjusted. ’148Pat., 12:29-34. For

example, the encoder can determine the actual QP to be used to quantize the DCT coefficients of a slice, as well as the difference (Δ QP) between the SQP and actual QP to be used for the slice in question. '148Pat., 20:28-40. The Δ QP is then transmitted in the bitstream (e.g., in the slice header). '148Pat., 20:40-46. A receiving decoder combines the Δ QP with a previously received SQP (e.g., in the sequence header for the video sequence) to determine the inverse level of quantization to be applied to the quantized DCT coefficients of the corresponding slice. '148Pat., 22:6-11, Fig. 7.

B. Brief Summary of '148 Patent Prosecution History

57. The '148 Patent claims priority to Provisional Application No. 60/374,667, filed April 23, 2002, and is a division of '629 application, filed April 23, 2003. '148Pat., (60), (62).

58. The '148 Patent's application was originally filed in conjunction with a preliminary amendment that canceled claims 1-3, 15, 16, and 28-36 and amended claims 4, 7-9, 12, 17, 18, 20, 26 and 27 of the original claims of the parent and added new claim 37. '148FH, 1-8. In particular, claims 4 and 17 were amended to be in independent form, and new claim 37 was an independent claim. '148FH, 4-9. Relevant to the '148 patent, independent claim 4 was generally related to a method of encoding, and independent claims 17 and 37 were generally related to video encoders. '148FH, 4, 6, and 8. Each of these independent claims recited *defining or selecting a default level of quantization*. '148FH, 4, 6, 8. Claims 4 and 17 further

recited *providing an indication of the said default level of quantization to a decoding process and in the encoded bitstream*, respectively. '148FH, 4, 6. Claim 37 further recited that *the default level is specific to the video coding applications*. '148FH, 8.

59. The Examiner rejected the claims in an Office Action dated November 23, 2010, under nonstatutory obviousness-type double patenting based on claims 1-10 of the '629 application. '148FH, 247-51. The Office Action lists prior art made of record and not relied upon that was considered pertinent to the Patent Owner's disclosure. '148FH, 251-52. The listed prior art does not include any of the prior art references relied upon in the petition. Patent Owner overcame the double patenting rejection by filing a Terminal Disclaimer on February 26, 2011. '148FH, 267, 331. The Examiner submitted a Notice of Allowance on January 3, 2012, and the '148 Patent was issued on May 8, 2012. '148FH, 332-33, 360.

60. Patent Owner further submitted a Request for Certificate of Correction to correct mistakes incurred through the fault of the USPTO on February 6, 2013. '148FH, 363. The Certificate of Correction substituted the title page of the '148 Patent to include a corrected illustrative figure and substituted Figures 5 and 6. '148FH, 364-66. The substituted Figures 5 and 6 were originally filed as part of the preliminary amendment filed in conjunction with the application on July 26, 2007. '148FH, 1-11.

IX. CLAIM CONSTRUCTION

61. I have given all the claim terms of the Challenged Claims their plain and ordinary meaning, as would have been understood by a POSITA, at the time of the alleged invention, unless otherwise discussed below.

A. Means-Plus-Function (MPF)

62. It is my understanding that when a claim limitation uses the term “means” or “module” and includes functional language, it is presumed to be a MPF term. *See* MPEP §2181. It is further my understanding that the presumption is overcome only if the limitation further includes the structure, material, or acts necessary to perform the recited function. *TriMed, Inc. v. Stryker Corp.*, 514 F.3d 1256, 1259-60 (Fed. Cir. 2008).

63. Claim 12 is directed to a *video encoder* and recites the following means-plus-function limitations: “*means for encoding...*,” “*means for transforming...*,” “*means for selecting...*,” and “*means for providing...*” Claim 23 is directed to a *video encoder* and recites the following means-plus-function limitations: “*a module for encoding...*,” “*a module for transforming...*,” and “*a module for selecting...*” The construction of each of these terms is addressed below.

1. “*Means for Encoding*” / “*Module for Encoding*”

64. The recited function of the means for encoding (claim 12) and the module for encoding (claim 23) 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.

65. Claim 12 recites:

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;

’148Pat., 26:51-54.

66. Claim 23 recites:

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;

’148Pat., 28:21-24.

67. The corresponding structure for the “means for encoding” and the “module for encoding” is the motion compensated predictor 150 and combiner 116 of the ’148 Patent.

68. With respect to the motion compensated predictor 150, the ’148 Patent explains:

Using the approximate representation of the motion vector generated by motion field coding block 140 and the pixel values of the best-matching candidate region from the reference frame, motion compensated prediction block 150 generates an array of predicted pixel values for each block of the macroblock.

’148Pat., 7:66-8:4.

69. In other words, the motion compensated predictor 150 (also described

as the motion compensated prediction block 150) 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. Thus, the motion compensated predictor applies motion compensated prediction to blocks of pixels.

70. With respect to the combiner 116, the ’148 Patent explains:

Each 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.

71. Further, in the ’148 Patent “[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. Thus, the combiner 116 provides blocks of prediction error values.

2. “Means for Transforming” / “Module for Transforming”

72. The function of the “means for transforming” (claim 12) and the “module for transforming” (claim 23) is “providing sets of transform coefficient values representative of said blocks of prediction error values.” ’148Pat., 26:55-57, 28:25-28.

73. Claim 12 recites:

means for transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values;

'148Pat., 26:55-57.

74. Claim 23 recites:

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;

'148Pat., 28:25-28.

75. The corresponding structure for the “means for transforming” and the “module for transforming” is the transform block 104.

76. With respect to the transform block 104, the '148 Patent explains:

Operation of video encoder 600 will now be considered in detail. When encoding a digital video sequence, encoder 600 operates in a manner similar to that previously described in connection with FIG. 3 to generate INTRA-coded and INTER-coded compressed video frames. As explained earlier in the text, in INTRA-coding mode, a Discrete Cosine Transform (DCT) is applied to each block of image data (pixel values) in order to produce a corresponding two-dimensional array of transform coefficient values. The DCT operation is performed in transform block 104 and the coefficients thus produced are subsequently passed to quantizer 106, where they are quantized. In INTER-coding mode, the DCT transform performed in block 104 is applied to blocks

of prediction error values. The transform coefficients produced as a result of this operation are also passed to quantizer 106 where they too are quantized.

'148Pat., 18:55-19:3.

77. In the '148 Patent, the transform block 104 (also described as the transform unit 104, DCT transformation block 104, and DCT transformer 104) applies a DCT transform to blocks of prediction error values to produce transform coefficients. '148Pat., 18:66-19:3. Thus, the DCT transform block 104 provides sets of transform coefficient values representative of blocks of prediction error values.

3. *“Means for Selecting” / “Module for Selecting”*

78. The function of the “means for selecting” (claim 12) is “selecting a default level of quantization for quantizing said sets of transform coefficient values.” '148Pat., 26:58-59. The function of the “module for selecting” (claim 23) is the same as the function of the “means for selecting,” with the additional element of “wherein the default level is specific to the video coding applications.” '148Pat., 28:29-32.

79. Claim 12 recites:

means for selecting a default level of quantization for quantizing said sets of transform coefficient values;

'148Pat., 26:58-59.

80. Claim 23 recites:

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.

'148Pat., 28:29-32.

81. The corresponding structure for the “means for selecting” and the “module for selecting” is the control manager 660.

82. With respect to the control manager 660, the '148 Patent explains:

The transform coefficients for each prediction error block are passed to quantizer 106 where they are quantized using a quantization parameter QP in a manner analogous to that described above in connection with operation of the encoder in INTRA-coding mode. Again, selection of the quantization parameter QP is controlled by the control manager 160 via control line 115.

'148Pat., 6:16-22.

Having determined the SQP value to be used for the sequence, control manager 660 provides an indication of the selected SQP value via control line 122 to video multiplex coder 670 which, in turn, inserts an indication of the SQP value into the bit-stream 635 of coded image information representative of the video sequence. Preferably, this indication is provided in a sequence header portion of the encoded video bit-stream 635.

'148Pat., 19:60-65.

83. In other words, the 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, the control manager 660 selects a default level of quantization for quantizing sets of transform coefficient values.

4. *“Means for Providing”*

84. The function of the “means for providing” (claim 12) is “providing an indication of said default level of quantization in the encoded bit-stream.” '148Pat., 26:60-62.

85. Claim 12 recites:

means for providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence.

'148Pat., 26:60-62.

86. The structure of the “means for providing” is the video multiplex coder 670.

87. With respect to the video multiplex coder 670, the '148 Patent explains:

Video multiplex coder 170 also receives motion vector information (described in the following) from motion field coding block 140 via line 126 and control information (e.g. including an indication of the quantization parameter QP) from control manager 160. It entropy codes the motion vector information and control information and forms a single bit stream of coded image information, 135 comprising the entropy coded motion vector, prediction error and control information. The indication, qz, of the quantization parameter QP is provided to multiplex [coder] 170 via line 124.

'148Pat., 6:38-47.

Having determined the SQP value to be used for the sequence, control manager 660 provides an indication of the selected SQP value via control line 122 to video multiplex coder 670 which, in turn, inserts an indication of the SQP value into the bit-stream 635 of coded image information representative of the video sequence. Preferably, this indication is provided in a sequence header portion of the encoded video bit-stream 635.

'148Pat., 19:60-65.

88. In other words, the video multiplex coder 670 encodes control information, such as quantization information, in a bit-stream. '148Pat., 6:38-47; 19:60-65. Thus, the video multiplex coder 670 provides an indication of a default level of quantization in an encoded bit-stream.

X. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Obviousness over MPEG-1

1. MPEG-1

89. MPEG-1 is titled "International Standard ISO/IEC 11172-2: Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbits/s – Part 2: Video." The first edition of MPEG-1 referenced herein (EX-1005) was ratified by the International Organization for Standardization on August 1, 1993, and I understand was made available to the public shortly after October 21, 1993. Munford, 9-11. I understand

that MPEG-1 is therefore prior art to the '148 Patent. MPEG-1 is a video compression standard that “specifies a coded representation that can be used for compressing video sequences to bitrates around 1.5 Mbit/s.” MPEG-1, 12. MPEG-1 describes a syntax for a coded video bitstream including six layers: a (1) sequence, (2) group-of-pictures, (3) picture, (4) slice, (5) macroblock, and (6) block. MPEG-1, 16.

90. MPEG-1 describes different picture types, including intra-coded pictures (I-pictures) and predictive coded pictures (P-pictures). MPEG-1, 13. I-pictures are coded without reference to other pictures while P-pictures are coded using motion compensated prediction from a past I-picture or P-picture. MPEG-1, 13. The coding uses motion compensation to reduce temporal redundancy in a video. MPEG-1, 13. The motion compensation used for P-pictures is also referred to as forward prediction as it forms a prediction from a past I-picture or P-picture (i.e., the last transmitted I or P-picture). MPEG-1, 96.

91. Motion compensation uses motion vectors, which give the displacement between picture elements (pels¹) in a previously transmitted picture and a predicted block. MPEG-1, 95. MPEG-1 discloses that “[t]he predicted block is usually a good estimate of the current block, and it is usually more efficient to

¹ Picture element (pel) is interchangeable with the term pixel.

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. MPEG-1 refers to the difference between a current block and a predicted block as a “prediction error.” MPEG-1, 13.

92. MPEG-1 describes how it “achieves a high compression ratio while preserving good picture quality,” by utilizing techniques that discard less important information (e.g., reduces the number of bits in the encoded video). MPEG-1, 12-13. For example, MPEG-1 describes compressing the 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. In step (1), the DCT transforms the prediction error into the DCT domain, thereby producing DCT coefficients. MPEG-1, 14-15. In step (2), quantization of the DCT coefficients produces quantized DCT coefficients that can be encoded using fewer bits. MPEG-1, 14, 69.

93. Quantization (step 2) is generally applied to represent a range of values by a single value in the range. In MPEG-1, each 8-pel by 8-line block of the prediction error for predictive-coded macroblocks is transformed into the DCT domain. MPEG-1, 14. Each resulting block of DCT coefficients is quantized. MPEG-1, 15.

94. Regarding quantization, MPEG-1 describes that:

Quantization represents a range of values by a single value in the range.
For example, converting a real number to the nearest integer is a

form of quantization. The quantized range can be concisely represented as an integer code, which can be used to recover the quantized value during decoding. The difference between the actual value and the quantized value is called the quantization noise. Under some circumstances, the HVS is less sensitive to quantization noise so such noise can be allowed to be large, thus increasing coding efficiency.

MPEG-1, 69 (emphasis added).

95. In other words, quantization compresses the video data by representing a range of values using a single value in the range, and an integer code representing the range “can be used to recover the quantized value during decoding.” MPEG-1, 69. Quantization of the DCT coefficients thereby produces quantized DCT coefficients that can be efficiently encoded using fewer bits (i.e., to reduce the number of the bits to be coded). MPEG-1, 14, 69.

96. MPEG-1 quantization uses a non-intra quantizer matrix (“non_intra_quantizer_matrix”) (red) in the sequence header syntax of a video sequence, as shown in the annotated sequence header syntax below.² MPEG-1, 36.

² MPEG-1 uses terms “quantizer matrix” and “quantization matrix” interchangeably. Additionally, MPEG-1 uses “non-intra quantization matrix” and “non_intra_quantizer_matrix” to refer to the same matrix. In formulas and/or pseudocode, MPEG-1 also uses “niqmatrix” and “non_intra_quant” to refer to the non-intra quantization matrix.

2.4.2.3 Sequence header

Syntax	No.of bits	Mnemonic
<code>sequence_header() {</code>		
<code>sequence_header_code</code>	32	bslbf
<code>horizontal_size</code>	12	uimsbf
<code>vertical_size</code>	12	uimsbf
<code>pel_aspect_ratio</code>	4	uimsbf
<code>picture_rate</code>	4	uimsbf
<code>bit_rate</code>	18	uimsbf
<code>marker_bit</code>	1	"1"
<code>vbv_buffer_size</code>	10	uimsbf
<code>constrained_parameters_flag</code>	1	
<code>load_intra_quantizer_matrix</code>	1	
if (load_intra_quantizer_matrix)		
<code>intra_quantizer_matrix []</code>	8*64	uimsbf
<code>load_non_intra_quantizer_matrix</code>	1	
if (load_non_intra_quantizer_matrix)		
<code>non_intra_quantizer_matrix []</code>	8*64	uimsbf
<code>next_start_code()</code>		
if (nextbits() == extension_start_code) {		
<code>extension_start_code</code>	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
<code>sequence_extension_data</code>	8	
}		
<code>next_start_code()</code>		
}		
if (nextbits() == user_data_start_code) {		
<code>user_data_start_code</code>	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
<code>user_data</code>	8	
}		
<code>next_start_code()</code>		
}		
}		

MPEG-1, 36 (annotated).

97. MPEG-1 defines this quantizer matrix by:

non_intra_quantizer_matrix -- This is a list of sixty-four 8-bit unsigned integers. The new values, stored in the zigzag scanning order shown in 2.4.4.1, replace the default values shown above. The value zero is forbidden. The new values shall be in effect until the next occurrence of a sequence header.

MPEG-1, 43.

98. The non-intra quantizer matrix (“non_intra_quantizer_matrix”) is thereby “a list of sixty-four 8-bit unsigned integers [that is] **in effect until the next occurrence of a sequence header.**”³ MPEG-1, 43 (emphasis added). Each unsigned integer of the matrix is used to quantize a corresponding DCT coefficient of the block of DCT coefficients where the block of DCT coefficients is produced from a block of the prediction error. MPEG-1, 53. For a further illustration of quantization, see my description of the figure illustrating quantization of an 8x8 block of DCT coefficients in §VII (Technology Overview).

99. MPEG-1 describes that quantization also involves a quantizer scale (“quantizer_scale”) (annotated red), provided in the slice layer syntax and/or the macroblock layer syntax, as shown in the annotated slice and macroblock layer syntaxes below. MPEG-1, 39.

³ The sixty-four integers represent an 8x8 matrix of integers, and correspond to an 8-pel by 8-line block.

Syntax	No. of bits	Mnemonic
slice() {		
slice_start_code	32	bslbf
quantizer_scale	5	uimsbf
while (nextbits() == '1') {		
extra_bit_slice	1	"1"
extra_information_slice	8	
}		
extra_bit_slice	1	"0"
do {		
macroblock()		
} while (nextbits() != '000 0000 0000 0000 0000 0000')		
next_start_code()		
}		

2.4.2.7 Macroblock layer

Syntax	No. of bits	Mnemonic
macroblock () {		
while (nextbits() == '0000 0001 111')		
macroblock_stuffing	11	vlc1bf
while (nextbits() == '0000 0001 000')		
macroblock_escape	11	vlc1bf
macroblock_address_increment	1-11	vlc1bf
macroblock_type	1-6	vlc1bf
if (macroblock_quant)		
quantizer_scale	5	uimsbf
if (macroblock_motion_forward) {		
motion_horizontal_forward_code	1-11	vlc1bf
if ((forward_f != 1) &&		
(motion_horizontal_forward_code != 0))		
motion_horizontal_forward_r	1-6	uimsbf
motion_vertical_forward_code	1-11	vlc1bf
if ((forward_f != 1) &&		
(motion_vertical_forward_code != 0))		
motion_vertical_forward_r	1-6	uimsbf
}		
if (macroblock_motion_backward) {		
motion_horizontal_backward_code	1-11	vlc1bf
if ((backward_f != 1) &&		
(motion_horizontal_backward_code != 0))		
motion_horizontal_backward_r	1-6	uimsbf
motion_vertical_backward_code	1-11	vlc1bf
if ((backward_f != 1) &&		
(motion_vertical_backward_code != 0))		
motion_vertical_backward_r	1-6	uimsbf
}		
if (macroblock_pattern)		
coded_block_pattern	3-9	vlc1bf
for (i=0; i<6; i++)		
block(i)		
if (picture_coding_type == 4)		
end_of_macroblock	1	"1"
}		

MPEG-1, 39 (annotated).

100. MPEG-1 defines the quantizer scale by:

quantizer_scale -- An unsigned integer in the range 1 to 31 used to scale the reconstruction level of the retrieved DCT coefficient levels. The

value zero is forbidden. The decoder shall use this value until another `quantizer_scale` is encountered either at the slice layer or the macroblock layer. The presence of quantizer-scale is determined from `macroblock_type`.⁴

MPEG-1, 46.

101. The quantizer scale is thereby an “unsigned integer in the range 1 to 31 used to scale the reconstruction level of the retrieved DCT coefficient levels.”

MPEG-1, 46. A POSITA would have understood that a “reconstruction level” refers to the outcome of inverse quantization (e.g., performed by a decoder) applied to a quantized coefficient (e.g., retrieved DCT levels), and that “scaling” the reconstruction level means multiplying the inverse quantization outcome by an appropriate quantizer scale factor. This means that `quantizer_scale` multiplies the outcome of inverse quantization (“reconstruction level”) applied to a quantized coefficient (“retrieved DCT coefficient level”).

102. MPEG-1 illustrates this in example code which teaches multiplying individual elements of the non-intra quantizer matrix by the quantizer scale on the decoding side. MPEG-1, 115, 53. As reflected in the annotated excerpt below describing the reconstruction of DCT coefficients on the decoding side, individual elements of the non-intra quantizer matrix (“`non_intra_quant`”) are multiplied by

⁴ Note that the `quantizer_scale` is also defined for the slice layer on MPEG-1, 45

the quantizer scale (“quantizer_scale”), and the result of that multiplication (i.e., the individual scaled quantization coefficients) is used to reconstruct DCT coefficients at the decoder (i.e., inverse quantization). MPEG-1, 53.

```

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 ;
    }
}

```

MPEG-1, 53 (annotated).

103. Similarly, MPEG-1 discloses that during encoding, each of the DCT coefficients is divided by a product corresponding to a multiplication of individual elements of the non-intra quantizer matrix by the quantizer scale. MPEG-1, 115. As shown in the annotated excerpt below, MPEG-1 describes that quantizing the DCT coefficients of a P-picture includes quantizing each coefficient by a corresponding element in the non-intra quantization matrix multiplied by the quantizer scale as described in (“coeffqant = (8*coefforig)/(quantscale*niqmatrix”). MPEG-1, 115. Each DCT coefficient “coefforig” is thereby divided by the product (*see* the denominator of the above listed formula clearly showing “quantscale*niqmatrix”) of a corresponding element in “niqmatrix” (referring to the

“non_intra_quantization_matrix” in the sequence header syntax) and the “quantscale” (referring to the “quantizer_scale” in the slice header syntax and macroblock header syntax).

```
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);
```

The process is illustrated below:

niqmatrix	16	16	16	16	16
quantscale	10	10	10	10	10
coefforig	-39~-20	-19~19	20~39	40~59	60~79
coeffqant	-1	0	1	2	3
coeffrec	-29	0	29	49	69

MPEG-1, 115 (annotated).

2. Claim 1 and Dependents

a. Claim 1

- (i) 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:

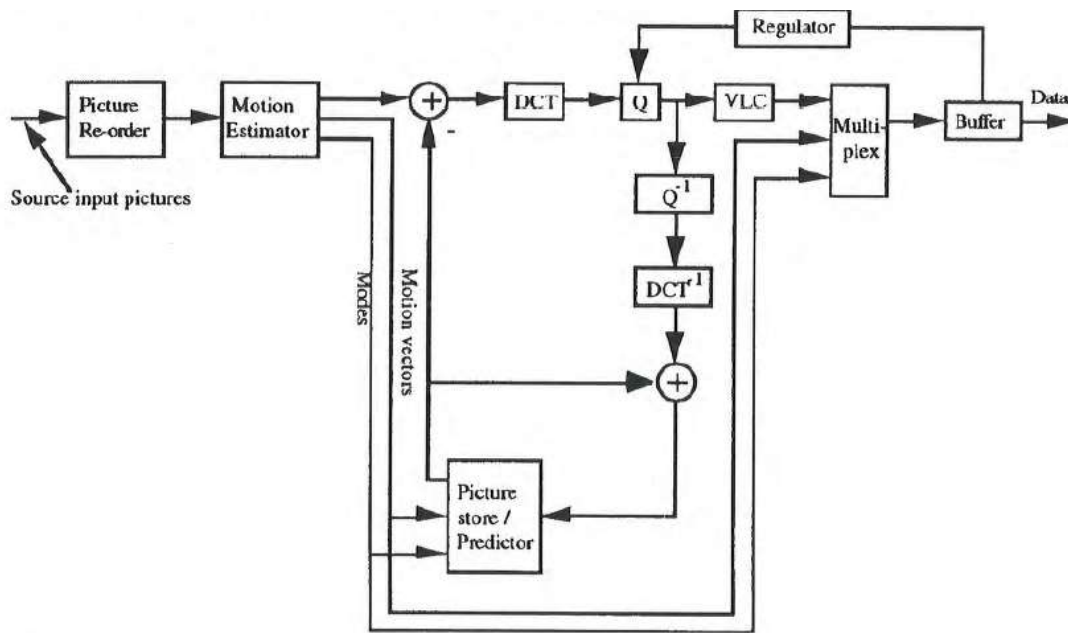
104. To the extent the preamble is limiting, MPEG-1 discloses or suggests 1[pre].

105. MPEG-1 is a standard defining a video coding process. MPEG-1, 12-

17. Figure 1 of MPEG-1 includes a block diagram of an encoder and describes functions performed by such an encoder 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. MPEG-1 therefore discloses or suggests *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* recited by 1[pre]. MPEG-1 specifies that “the syntax and semantics of the bitstream and the signal processing” are used in encoding video sequences. MPEG-1, 14.

Additionally, Annex D of MPEG-1 explains coding methods related to the normative decoding clauses of MPEG-1. MPEG-1, 68.



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

MPEG-1, 14.

106. MPEG-1 also discloses a syntax for a coded video bitstream including six layers: a (1) sequence; (2) group-of-pictures; (3) picture; (4) slice; (5) macroblock; and (6) block. MPEG-1, 16. MPEG-1 describes that “the term ‘picture’ is always used...in preference to the terms field or frame.” MPEG-1, 25. According to MPEG-1, a picture is therefore “*a frame.*” A video sequence is a series of one or more groups of pictures. MPEG-1, 27. MPEG-1 therefore discloses or suggests *the digital video sequence comprising a number of frames*, recited by 1[pre]. A group of pictures is a series of one or more coded pictures. MPEG-1, 32. This discloses or suggests that *each frame of said sequence comprising an array of pixels*, as recited

by 1[pre]. Further, each picture consists of one or more slices. MPEG-1, 71. The slices of a picture do not overlap and there are no gaps between slices. MPEG-1, 34.

107. Each slice includes at least one macroblock. MPEG-1, 34. An arrangement of blocks in a macroblock is illustrated in Figure 6 of MPEG-1 below. As shown, each macroblock has 4 luminance blocks (representing brightness) and 2 chrominance blocks (representing color difference). MPEG-1, 34. Each macroblock is therefore *divided into a plurality of blocks*, as is recited in 1[pre]. MPEG-1, 34. Further, each block (whether a luminance or chrominance component) represents an 8-pel by 8-line section meaning that *each block comprises a certain number of said pixels*, recited by 1[pre]. MPEG-1, 34. Each macroblock contains a 16-pel by 16-line section of luminance (Y) (since it contains 4 arranged 8-pel by 8-line luminance blocks, indicated with numbers 0-3 in Figure 6 below) and spatially corresponding 8-pel by 8-line section of each chrominance component (Cb and Cr) (indicated with numbers 4 and 5 below). MPEG-1, 34. MPEG-1 pictures thus comprise pixels, where the pixels are divided into “*blocks*” (e.g., luminance and chrominance blocks).

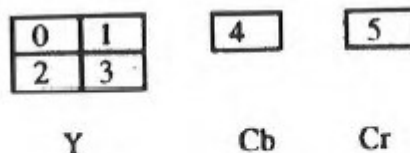


Figure 6 -- The arrangement of blocks in a macroblock.

MPEG-1, 34.

- (ii) **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;**

108. MPEG-1 discloses or suggests 1[a].

109. With regard to temporal processing, MPEG-1 describes using temporal processing of P-pictures:

Because of the conflicting requirements of random access and highly efficient compression, three main picture types are defined. Intra-coded pictures (I-Pictures) are coded without reference to other pictures. They provide access points to the coded sequence where decoding can begin, but are coded with only a moderate compression ratio. Predictive coded pictures (P-Pictures) are coded more efficiently using motion compensated prediction from a past intra or predictive coded picture and are generally used as a reference for further prediction. Bidirectionally-predictive coded pictures (B-Pictures) provide the highest degree of compression but require both past and future reference pictures for motion compensation. Bidirectionally-predictive coded pictures are never used as references for prediction. The organisation of the three picture types in a sequence is very flexible. The choice is left to the encoder and will depend on the requirements of the application.

MPEG-1, 13.

110. In other words, MPEG-1 discloses *encoding* that uses temporal processing of P-pictures “using motion compensated prediction from a past intra or predictive coded picture.” MPEG-1, 13. MPEG-1 therefore discloses or suggests

encoding a frame of the digital video sequence by applying motion compensated prediction, recited by 1[a]. As I explained in §X.A.1 (MPEG-1 overview), P-pictures use forward motion compensation from the past (i.e., the last transmitted) I- or P-picture. MPEG-1, 96.

111. MPEG-1 further describes motion compensation and interframe coding by:

Motion compensation (MC) predicts the values of a block pels in a picture by relocating a block of neighbouring pel values from a known picture. The motion is described as a two-dimensional motion vector that specifies where to retrieve a block of pel values from a previously decoded picture that is used to predict pel values of the current block. The simplest example is a scene where the camera is not moving, and no objects in the scene are moving. The pel values at each image location remain the same, and the motion vector for each block is 0. In general however, the encoder may transmit a motion vector for each macroblock. The translated block from the known picture becomes a prediction for the block in the picture to be encoded. The technique relies on the fact that within a short sequence of pictures of the same general scene, many objects remain in the same location while others move only a short distance.

MPEG-1, 70.

112. Motion compensation thereby “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. MPEG-1 describes that the prediction uses motion vectors

that provide offsets into the past I- or P-picture used as a reference containing the decoded pel values. MPEG-1, 24. These offsets are used to form a prediction error. MPEG-1, 24. One motion vector is used for each non-intra macroblock in P-pictures. MPEG-1, 14.

113. As I explained in §X.A.2.a(i) (for 1[pre]), MPEG-1 discloses that a macroblock includes four blocks of luminance component and two blocks of chrominance component. MPEG-1, 15, 34. Each block (whether representing the luminance or chrominance component) is an 8-pel by 8-line section. MPEG-1, 15, 34. MPEG-1 thereby discloses or suggests a *block comprising a certain number of said pixels* or a *block of pixels*, as recited by 1[a]. The one motion vector used for each macroblock—including the chrominance and luminance blocks therein—is therefore applied to *blocks of pixels*. MPEG-1, 13-14. Thus, by describing that a motion vector is applied for each non-intra macroblock, MPEG-1 *applies motion compensated prediction to blocks of pixels*, recited by 1[a]. And the motion compensated prediction uses those motion vectors to form prediction error signals for the respective blocks. MPEG-1, 24. This discloses or suggests *producing corresponding blocks of prediction error values*, recited by 1[a].

- (iii) **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;**

114. MPEG-1 discloses or suggests 1[b].

115. MPEG-1 discloses using DCT, which is a transform coding technique, to transform the prediction error signals. MPEG-1, 13. MPEG-1 thereby discloses or suggests *applying a transform coding technique to said blocks of prediction error values*, recited by 1[b].

116. With respect to the transformation of predictive-coded blocks, MPEG-1 describes:

Motion compensation is used for causal prediction of the current picture from a previous picture, for non-causal prediction of the current picture from a future picture, or for interpolative prediction from past and future pictures. Motion vectors are defined for each 16-pel by 16-line region of the picture. The difference signal, the prediction error, is further compressed using the discrete cosine transform (DCT) to remove spatial correlation before it is quantized in an irreversible process that discards the less important information. Finally, the motion vectors are combined with the DCT information, and coded using variable length codes.

MPEG-1, 13.

Firstly, for a given macroblock, the coding mode is chosen. It depends on the picture type, the effectiveness of motion compensated prediction in that local region, and the nature of the signal within the block. Secondly, depending on the coding mode, a motion compensated

prediction of the contents of the block based on past and/or future reference pictures is formed. This prediction is subtracted from the actual data in the current macroblock to form an error signal. Thirdly, this error signal is separated into 8 by 8 blocks (4 luminance and 2 chrominance blocks in each macroblock) and a discrete cosine transform is performed on each block. Each resulting 8 by 8 block of DCT coefficients is quantized and the two-dimensional block is scanned in a zig-zag order to convert it into a one-dimensional string of quantized OCT coefficients. Fourthly, the side-information for the macroblock (mode, motion vectors etc) and the quantized coefficient data are encoded. For maximum efficiency, a number of variable length code tables are defined for the different data elements. Run-length coding is used for the quantized coefficient data.

MPEG-1, 15.

117. In other words, specifically, each 8-pel by 8-line block of a predictive-coded macroblock—where each block represents prediction error—is transformed using the same DCT. MPEG-1, 13-15 (“Thirdly, this error signal is separated into 8 by 8 blocks (4 luminance and 2 chrominance blocks in each macroblock) and a discrete cosine transform is performed on each block”). MPEG-1 further discloses that the difference signal between a motion compensated macroblock and the current macroblock is transformed with a two-dimensional DCT to produce an 8-pel by 8-line array of transform coefficients. MPEG-1, 110. MPEG-1 therefore discloses or suggests *produc[ing] sets of transform coefficient values representative of said blocks of prediction error values*, recited by 1[b]. MPEG-1, 110.

- (iv) **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**

118. MPEG-1 discloses or suggests 1[c].

119. MPEG-1 describes, *inter alia*, encoding including quantization:

Firstly, for a given macroblock, the coding mode is chosen. It depends on the picture type, the effectiveness of motion compensated prediction in that local region, and the nature of the signal within the block. Secondly, depending on the coding mode, a motion compensated prediction of the contents of the block based on past and/or future reference pictures is formed. This prediction is subtracted from the actual data in the current macroblock to form an error signal. Thirdly, this error signal is separated into 8 by 8 blocks (4 luminance and 2 chrominance blocks in each macroblock) and a discrete cosine transform is performed on each block. Each resulting 8 by 8 block of DCT coefficients is quantized and the two-dimensional block is scanned in a zig-zag order to convert it into a one-dimensional string of quantized DCT coefficients. Fourthly, the side-information for the macroblock (mode, motion vectors, etc) and the quantized coefficient data are encoded. For maximum efficiency, a number of variable length code tables are defined for the different data elements. Run-length coding is used for the quantized coefficient data.

MPEG-1, 15.

120. In other words, MPEG-1 discloses that each 8-pel by 8-line “block of DCT coefficients [is quantized] and scanned in a zig-zag order to convert it into a

one-dimensional string of quantized DCT coefficients.” MPEG-1, 15. The “block of DCT coefficients” corresponds to *the sets of transform coefficient values* and MPEG-1 thereby discloses or suggests quantizing *the sets of transform coefficient values*, recited in 1[c]. With respect to encoding of sequences, MPEG-1 discloses a quantizer matrix for quantizing DCT coefficients thereby disclosing or suggesting *to quantize the sets of transform coefficient values*, recited by 1[c]. MPEG-1, 83. The details of how this quantization is accomplished had been described earlier (*see* X.A.2.a(i)).

121. MPEG-1 discloses the following sequence header syntax:

2.4.2.3 Sequence header

Syntax	No.of bits	Mnemonic
<code>sequence_header() {</code>		
<code>sequence_header_code</code>	32	bslbf
<code>horizontal_size</code>	12	uimsbf
<code>vertical_size</code>	12	uimsbf
<code>pel_aspect_ratio</code>	4	uimsbf
<code>picture_rate</code>	4	uimsbf
<code>bit_rate</code>	18	uimsbf
<code>marker_bit</code>	1	"1"
<code>vhv_buffer_size</code>	10	uimsbf
<code>constrained_parameters_flag</code>	1	
<code>load_intra_quantizer_matrix</code>	1	
if (load_intra_quantizer_matrix)		
<code>intra_quantizer_matrix []</code>	8*64	uimsbf
<code>load_non_intra_quantizer_matrix</code>	1	
if (load_non_intra_quantizer_matrix)		
<code>non_intra_quantizer_matrix []</code>	8*64	uimsbf
<code>next_start_code()</code>		
if (nextbits() == extension_start_code) {		
<code>extension_start_code</code>	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
<code>sequence_extension_data</code>	8	
}		
<code>next_start_code()</code>		
}		
if (nextbits() == user_data_start_code) {		
<code>user_data_start_code</code>	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
<code>user_data</code>	8	
}		
<code>next_start_code()</code>		
}		
}		

MPEG-1, 36 (annotated).

122. MPEG-1 defines the parameter “load_non_intra_quantizer_matrix” (red) as:

load_non_intra_quantizer_matrix -- This is a one-bit flag which is set to "1" if a non_intra_quantizer_matrix follows. If it is set to "0" then the default values defined below are used until the next occurrence of the sequence header.

16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16

MPEG-1, 42.

123. In other words, the parameter “load_non_intra_quantizer_matrix” (red) in the sequence header syntax is a one-bit flag that indicates how the non-intra quantization matrix is specified. MPEG-1, 36, 42. The non-intra quantization matrix is used to quantize the DCT coefficients as described earlier (*see* X.A.2.a(i)). Specifically, when the flag is set to “0,” the non-intra quantizer matrix is “reset” to “default values” which is an 8-pel by 8-line matrix with all its values equal to 16 (as shown above). MPEG-1, 42, 83-84. MPEG-1 further describes that this 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 (emphasis added).

124. Alternatively, when the “load_non_intra_quantizer_matrix” (red) flag is set to “1,” the flag is followed in the sequence header by “a list of sixty-four 8-bit unsigned integers,” (green) which is used in place of the “default” values of all 16s (i.e., used when the flag is set to “0”). MPEG-1, 36, 42-43, 83. MPEG-1 further discloses that these values are in effect **until the next occurrence of a sequence header.** MPEG-1, 43.

125. 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 in the sequence header) or to the values provided in the sequence header (i.e., “load_non_intra_quantizer_matrix” is set to “1” in the sequence header), the non-intra quantizer matrix provides *a default level of quantization*, as recited in 1[c]. MPEG-1, 42, 43, 83.

126. Moreover, MPEG-1 describes that the non-intra quantizer matrix may be updated during video encoding with a later occurrence of a sequence header. This behavior is consistent with the ’148 Patent. With respect to the default level of quantization, the ’148 Patent recites:

Advantageously, the default level of quantization to be used throughout encoding of the digital video sequence to quantize the sets of transform coefficient values can be updated during encoding of the digital video sequence and a further indication indicative of the updated default level of quantization is provided.

’148Pat., 12:19-24.

127. In other words, the ’148 Patent explains that 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. That is, the default level of quantization can change, and MPEG-1 provides the same.

128. Given the ’148 Patent’s explanation that a default level of quantization is used for encoding until it is updated (’148Pat., 12:19-24), a POSITA would have

understood that “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 (see discussion at ¶ 121). A POSITA would have understood that when an MPEG-1 encoder determines the value of the “load_non_intra_quantizer_matrix” flag (e.g., to specify whether the non-intra quantizer matrix is to be reset to the default values of all 16s, or whether the matrix will utilize the values provided in the sequence header), the encoder is selecting or determining a level of quantization. Similarly, a POSITA would have understood that when an MPEG-1 encoder sets “load_non_intra_quantizer_matrix” to “1,” the values that follow in the sequence header (used for the non-intra quantizer matrix) are levels of quantization selected or determined by the encoder. In either case, a POSITA would understand MPEG-1 to *define a default level of quantization for use in encoding of the digital video sequence*, as recited in 1[c].

(v) 1[d]: providing an indication of the default level of quantization to a decoding process.

129. MPEG-1 discloses or suggests 1[d].

130. The sequence header of MPEG-1, as I described in §X.A.2.a(iv) (for 1[c]), is a block of data in the coded bitstream that includes a representation of data elements for decoding a video sequence. MPEG-1, 26, 32. The sequence header provides information to the decoder to decode and playback a video sequence.

Parameters defining the quantizer matrices may be redefined each time a sequence header occurs in the bitstream. MPEG-1, 32. The sequence header of MPEG-1 therefore teaches *providing...to a decoding process*, recited in 1[d].

131. As I explained in X.A.2.a(iv) (for 1[c]), MPEG-1 discloses a flag (“load_non_intra_quantizer_matrix”) in the sequence header that indicates how values for the non-intra quantizer matrix are to be specified. MPEG-1, 36, 42. A POSITA would have understood that the flag itself is *an indication of the default level of quantization*, recited in 1[d]. To the extent Patent Owner argues that a flag does not satisfy this limitation, however, MPEG-1 further discloses that when the “load_non_intra_quantizer_matrix” flag is set to “1,” the flag is followed in the sequence header by “a list of sixty-four 8-bit unsigned integers.” MPEG-1, 36, 42-43, 83. This list of sixty-four 8-bit unsigned integers is used as a non-intra quantizer matrix in place of the “default” values of all 16s. MPEG-1, 42-43. MPEG-1 therefore discloses that the sequence header includes in the bitstream, when the flag is set to “1,” the values of the non-intra quantizer matrix, which additionally discloses or suggests *an indication of the default level of quantization*. Further, a decoder will decode the sequence header, including sequence parameters. MPEG-1, 120.

- b. Claim 2: An encoding method according to claim 1, wherein said indication of the default level of quantization is provided in the encoded bit-stream representative of the digital video sequence.**

132. MPEG-1 discloses or suggests claim 2's additional limitation.

133. As I described in §X.A.2.a(v) (for 1[d]), MPEG-1 discloses an indication of the quantizer matrix (e.g., non-intra quantizer matrix) provided in a sequence header. MPEG-1, 36, 42, 43. This quantizer matrix of MPEG-1 discloses *the default level of quantization*, recited in claim 2. MPEG-1 teaches that the sequence header begins a coded video sequence. MPEG-1, 32. MPEG-1 thereby discloses or suggests *providing the default level of quantization in the encoded bit-stream representative of the digital video sequence*, recited in claim 2.

- c. Claim 3: An encoding method according to claim 2, wherein the encoded bit-stream is transmitted from a video encoding device to a corresponding video decoding device.**

134. MPEG-1 discloses or suggests claim 3's additional limitation.

135. MPEG-1 is a video compression standard describing the syntax and semantics of an encoded bitstream and the signal processing in the decoder. MPEG-1, 26, 32. 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. MPEG-1, 14, 16. A POSITA would have understood that MPEG-1 discloses or suggests the transmission of encoded video from a video encoding device to a corresponding

video decoding device. MPEG-1 therefore discloses or suggests an encoding method where *the encoded bit-stream is transmitted from...encoding device to a corresponding...decoding device*, recited in claim 3.

- d. Claim 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.**

136. MPEG-1 discloses or suggests claim 4's additional limitation.

137. As I described in §X.A.2.a(v) (for 1[d]), the non-intra quantizer matrix of MPEG-1 indicates *the default level of quantization* in the sequence header.

MPEG-1, 42, 43. MPEG-1 describes that a coded video sequence commences with a sequence header followed by one or more groups of pictures, and immediately before each of the groups of pictures, there may be a sequence header. MPEG-1, 32.

138. MPEG-1 further describes that once used, the non-intra quantizer matrix is used until the next occurrence of the sequence header. MPEG-1, 42, 43. The non-intra quantizer matrix can then be redefined when another sequence header occurs in the bitstream. MPEG-1, 32. MPEG-1 therefore discloses or suggests *the default level of quantization is updated*, recited in claim 4. For example, the flag `load_non_intra_quantizer_matrix` may be set, in a later sequence header, to a value different from how it had been set in a prior sequence header (such that one sequence header specifies the use of the “default values” of all 16s, and the other

sequence header specifies the use of the matrix values that follow in the bitstream). As a further example, `load_non_intra_quantizer_matrix` may be set to “1” in a later sequence header, followed by matrix values that differ from those of the non-intra quantizer matrix indicated by the prior sequence header. Thus, the non-intra quantizer matrix can be updated during the encoding by a new occurrence of a sequence header which discloses or suggests *providing an indication of the updated default level of quantization*, recited in claim 4.

- e. **Claim 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.**

139. MPEG-1 discloses or suggests claim 5’s additional limitation.

140. As I described in §X.A.2.a(iv) for (1[c]), 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.

141. MPEG-1 further describes providing a quantizer scale (“`quantizer_scale`”) (red) parameter in the slice header syntax, and optionally in the macroblock header syntax, as shown in the annotated excerpt below. MPEG-1, 39.

Syntax	No. of bits	Mnemonic
slice() {		
slice_start_code	32	bslbf
quantizer_scale	5	uimsbf
while (nextbits() == '1') {		
extra_bit_slice	1	"1"
extra_information_slice	8	
}		
extra_bit_slice	1	"0"
do {		
macroblock()		
} while (nextbits() != '000 0000 0000 0000 0000 0000')		
next_start_code()		
}		

2.4.2.7 Macroblock layer

Syntax	No. of bits	Mnemonic
macroblock () {		
while (nextbits() == '0000 0001 111')		
macroblock_stuffing	11	vlc1bf
while (nextbits() == '0000 0001 000')		
macroblock_escape	11	vlc1bf
macroblock_address_increment	1-11	vlc1bf
macroblock_type	1-6	vlc1bf
if (macroblock_quant)		
quantizer_scale	5	uimsbf
if (macroblock_motion_forward) {		
motion_horizontal_forward_code	1-11	vlc1bf
if ((forward_f != 1) &&		
(motion_horizontal_forward_code != 0))		
motion_horizontal_forward_r	1-6	uimsbf
motion_vertical_forward_code	1-11	vlc1bf
if ((forward_f != 1) &&		
(motion_vertical_forward_code != 0))		
motion_vertical_forward_r	1-6	uimsbf
}		
if (macroblock_motion_backward) {		
motion_horizontal_backward_code	1-11	vlc1bf
if ((backward_f != 1) &&		
(motion_horizontal_backward_code != 0))		
motion_horizontal_backward_r	1-6	uimsbf
motion_vertical_backward_code	1-11	vlc1bf
if ((backward_f != 1) &&		
(motion_vertical_backward_code != 0))		
motion_vertical_backward_r	1-6	uimsbf
}		
if (macroblock_pattern)		
coded_block_pattern	3-9	vlc1bf
for (i=0; i<6; i++)		
block(i)		
if (picture_coding_type == 4)		
end_of_macroblock	1	"1"
}		

MPEG-1, 39 (annotated).

142. MPEG-1 defines the quantizer scale by:

quantizer_scale -- An unsigned integer in the range 1 to 31 used to scale the reconstruction level of the retrieved DCT coefficient levels. The

value zero is forbidden. The decoder shall use this value until another quantizer_scale is encountered either at the slice layer or the macroblock layer. The presence of quantizer-scale is determined from macroblock_type.

MPEG-1, 46.

143. The quantizer scale is, in other words, an unsigned integer in the range 1 to 31, used to scale the reconstruction level of DCT coefficients until another quantizer scale is provided (e.g., in the slice layer or the macroblock layer), where the provided value is used until another quantizer scale is provided at the slice layer or the macroblock layer. MPEG-1, 45-46. That is, the quantizer scale multiplicatively scales the quantization values specified in the non-intra quantizer matrix. MPEG-1, 43. When the quantizer scale is set to “1,” the quantizer scale has no effect on the non-intra quantizer matrix set in the sequence header (*the default level of quantization*). When the quantizer scale is set to be an integer different from “1,” the quantization is performed according to the scaled quantization matrix. When the quantizer scale is an integer different from “1,” *a level of quantization [is] different from the default level of quantization*, as disclosed by MPEG-1.

144. MPEG-1 further discloses that the quantization of macroblocks in P-pictures depends on macroblock type. MPEG-1, 110-11. The macroblock types are listed in Table D.19 below. MPEG-1 explains that when the QUANT-column in table D.19 has a 1, a quantizer scale is transmitted. MPEG-1, 110-11. Table D.19 is

annotated to highlight that macroblock types “pred-mcq” and “pred-cq” therefore include the transmission of a quantizer scale (since “QUANT” is set to 1). MPEG-1, 110-11. These two macroblock types apply to motion compensated, coded non-intra macroblocks. MPEG-1, 112. In contrast, when the macroblock type is “pred-mc” (blue), the quantizer scale is not transmitted. MPEG-1, 110-12. And as explained below, in such instances the quantization is done using a previously established quantization level. MPEG-1, 110-12.

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				

MPEG-1, 110 (annotated).

145. For quantization of P-pictures applying the quantizer scale, MPEG-1 discloses that non-intra macroblocks are quantized using the non-intra quantization matrix and the quantizer scale. MPEG-1, 115. Regardless of whether the quantizer scale is provided at the slice level or at the macroblock level, as I described above, the quantizer scale (also referred to as “quantscale”) is used to modify the non-intra quantization matrix (also referred to as “niqmatrix”). MPEG-1, 115. As shown in the annotated excerpt below, MPEG-1 discloses the modification in accordance

with the equation “coeffqant = (8* coefforig)/(quantscale * niqmatrix),” where “coeffqant” is the quantized coefficient and “coefforig” is the original DCT coefficient. MPEG-1, 115.

```
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);
```

MPEG-1, 115.

146. Also, as I explained in §X.A.1 (MPEG-1 overview), MPEG-1 teaches multiplying 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,

(“dct_recon[m][n]=(((2*dct_zz[i]+Sign(dct_zz[i]*quantizer_scale*non_intra_quant[m][n])/16”). MPEG-1, 53.

```
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 ;
    }
}
```

MPEG-1, 53.

147. Given that the DCT coefficients of the non-intra macroblocks of P-

pictures are quantized using non-intra quantization matrix multiplied by the quantizer scale using the above equation, the quantized coefficients are quantized with a level of quantization that is *different from the default level of quantization* (i.e., the non-intra quantizer matrix), as is recited in claim 5.

f. Claim 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.

148. MPEG-1 discloses or suggests claim 6's additional limitation.

149. As I explained in §X.A.2.e (for claim 5), MPEG-1 discloses that the quantized coefficients are quantized with a level of quantization that is different from the non-intra quantization matrix by disclosing the quantizer scale that scales the non-intra quantization matrix. MPEG-1, 45. MPEG-1 also discloses that the quantizer scale is provided in the slice header syntax or in the macroblock header syntax. MPEG-1, 45, 46. The slice header includes data that is needed to decode a slice in a video frame. MPEG-1 further describes that the quantizer scale is used by the decoder in decoding until another quantizer scale is encountered either at the slice layer or the macroblock layer. MPEG-1, 45, 46. 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. MPEG-1 thereby discloses or suggests *an indication of a difference between said level of quantization and the default level of quantization*, recited in claim 6.

150. A POSITA would understand that the term “difference” encompasses a quality or state of something that is distinct or dissimilar from something else. Accordingly, an “indication of a difference between said level of quantization and the default level of quantization” encompasses an indication that these two levels of quantization are distinct or dissimilar. The quantizer scale that changes the non-intra quantization matrix to be used in encoding of the subsequent slices is such an indication of a difference between the non-intra quantization matrix and the scaled non-intra quantization matrix. MPEG-1 therefore discloses or suggests *a difference*, in the context of claim 6.

151. To the extent that Patent Owner asserts that this claim requires an arithmetic difference between *said level of quantization and the default level of quantization*, the quantizer scale of MPEG-1 additionally discloses or suggests an indication of a difference.

152. The '148 Patent explains:

Advantageously, the level of inverse quantization applied to the sets of quantized transform coefficient values can be adjusted such that an actual level of inverse quantization applied to the sets of quantized transform coefficient values is different from the default level of inverse quantization to be used throughout decoding of the encoded digital video sequence. In this case, the actual level of inverse quantization is determined by adding a difference value to the default level of inverse quantization, the difference value being representative of the difference between the default level of inverse quantization and the actual level of

inverse quantization to be applied. Preferably, an indication of the difference value is retrieved from a bit-stream representative of the encoded digital video sequence.

'148Pat., 13:57-41:3.

153. In other words, the '148 Patent explains that 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. As I described in §X.A.2.e (for claim 5), MPEG-1's quantizer scale, which is *an indication of a difference*, specifies the amount by which values of the non-intra quantizer matrix are multiplied. As I explained in §X.A.2.a(iv) (for 1[c]), the non-intra quantizer matrix corresponds to the *default level of quantization*. Further, a POSITA would have understood that multiplication indicates a repeated addition. In other words, the quantizer scale indicates the number of times a value of the non-intra quantizer matrix is added to itself. For example, when the quantizer scale is 1, the value is not added to itself; when the quantizer scale is 2, the value is added to itself once. Since the quantizer scale 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*), the quantizer scale discloses or suggests *an indication of a difference between said level of quantization and the default level of quantization*, recited in claim 6.

- g. Claim 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.**

154. MPEG-1 discloses or suggests claim 7's additional limitation.

155. As I described in §X.A.2.e (for claim 5), MPEG-1 describes providing a quantizer scale ("quantizer_scale") parameter 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 level of quantization (i.e., the non-intra quantizer matrix as I described in §X.A.2.a(iv) for 1[c]). MPEG-1, 42-43.

MPEG-1 thereby discloses or suggests *an actual level of quantization...is different from the default level of quantization.*

156. Moreover, MPEG-1 describes that the quantizer scale is used to scale the DCT coefficients until another quantizer scale is specified. MPEG-1, 46. A POSITA would therefore have understood that 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*, as recited in claim 7.

157. As an example of a scenario where the quantizer scale is adjusted from one picture to another, MPEG-1 describes that "[p]ictures consist of a header and

one or more slices.” MPEG-1, 71. A picture may therefore include only a single slice. For example, a picture of a uniform, static background (e.g., a white uniform-looking wall) can consist of a single slice. 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.

h. Claim 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.

158. MPEG-1 discloses or suggests claim 8’s additional limitation.

159. As I described in §X.A.2.f (for claim 6), the MPEG-1’s quantizer scale indicates a difference with respect to the non-intra quantizer matrix (i.e., *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 (i.e., *said actual level of quantization*).

i. Claim 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.

160. MPEG-1 discloses or suggests claim 9’s additional limitation.

161. As I described in §X.A.2.f (for claim 6), the quantizer scale indicates a difference between the level of quantization indicated by the non-intra quantizer matrix (i.e., *the default level of quantization*, discussed in §X.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 (i.e., *said actual level of quantization*). And as further described in §X.A.2.f for 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. MPEG-1 thereby discloses or suggests *providing to a decoding process an indication of a difference between said actual level of quantization and the default level of quantization*, recited in claim 9.

- j. **Claim 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.**

162. MPEG-1 discloses or suggests claim 10's additional limitation.

163. MPEG-1 describes a slice as a series of macroblocks, each macroblock 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. Macroblocks are made up of four luminance blocks and two chrominance blocks. MPEG-1, 24. A slice is formed by one or more macroblocks. MPEG-1, 34. According to MPEG-1, both a macroblock and a slice thereby correspond to *a plurality of blocks of pixels ... grouped into one or more segments*, recited in claim 10.

164. MPEG-1's luminance and chrominance blocks disclose *plurality of blocks of pixels, into which a frame of said digital video sequence is divided* for the reasons as I described in §X.A.2.a(i) (for 1[pre]) and the substantially similar limitation discussed therein. 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. Each macroblock of MPEG-1 therefore discloses or suggests *a segment into which a plurality of blocks of pixels* (e.g., four luminance blocks and two chrominance blocks) *are grouped*, as recited in claim 10.

165. Further, as I described in §X.A.2.e (for claim 5), MPEG-1's quantizer scale is used to scale the non-intra quantizer matrix. MPEG-1, 115. The quantizer scale ("quantizer_scale") parameter is provided in the slice header syntax that appears in each slice. MPEG-1, 39, 91. Additionally, the quantizer scale parameter is also provided in the macroblock level syntax of each macroblock. MPEG-1, 39, 72. MPEG-1's non-intra quantizer matrix is thus scaled slice-by-slice or macroblock-by-macroblock, either of which illustrates *blocks of pixels...grouped into one or more segments*. Therefore, MPEG-1's quantizer scale discloses or 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*, recited in claim 10.

- k. **Claim 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.**

166. MPEG-1 discloses or suggests claim 11's additional limitation.

167. As I described in §X.A.2.j (for claim 10), the quantizer scale is provided in the slice header syntax at the start of each slice or in the macroblock header syntax at the start of each macroblock to the decoding process to indicate which level of quantization is to be used. MPEG-1, 39, 91. Further, as I described in §X.A.2.f (for claim 6), 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). MPEG-1 thereby discloses or suggests, *providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization*, recited in claim 11.

3. **Claim 12 and Dependents**

a. **Claim 12**

- (i) **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:**

168. If the preamble is limiting, MPEG-1 discloses or suggests 12[pre].

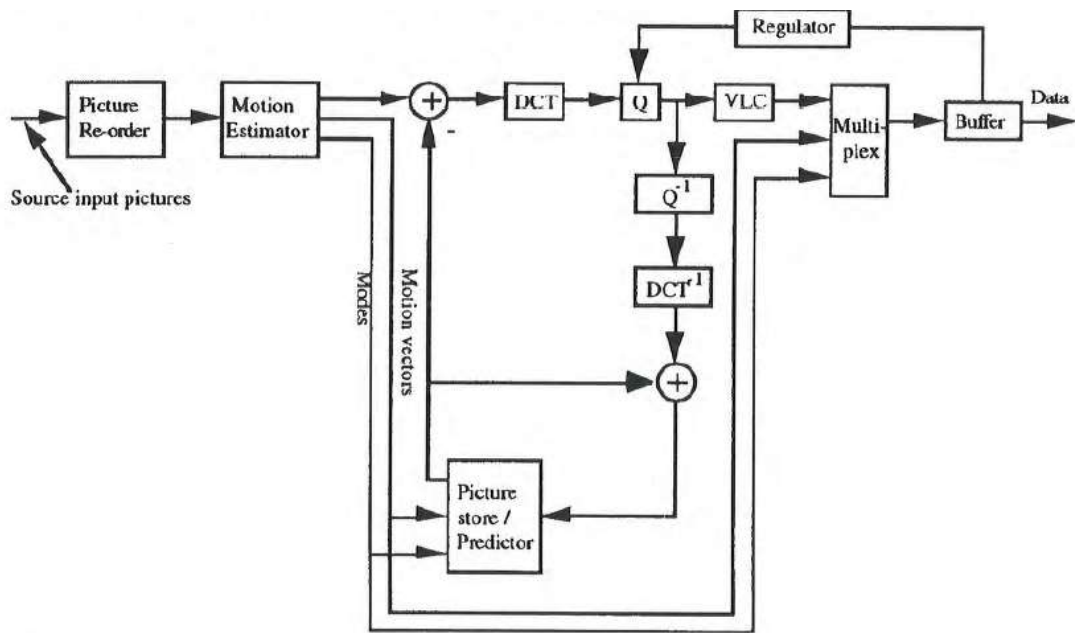
169. As I described in §X.A.2.a(i) (for 1[pre]), MPEG-1 discloses a method *to produce an encoded video bit-stream representative of the digital video sequence, the digital video sequence comprising a number of frames, each frame of said sequence comprising an array of pixels divided into a plurality of blocks, each block comprising a certain number of said pixels*, recited in 12[pre].

170. Further, Figure 2 of MPEG-1 (reproduced below) describes an encoder. MPEG-1, 14. MPEG-1 describes encoding by:

2.1.57 encoding (process): A process, not specified in ISO/IEC 11172, that reads a stream of input pictures or audio samples and produces a valid coded bitstream as defined in ISO/IEC 11172.

MPEG-1, 22.

171. MPEG-1 also describes that the operations of the encoder can include producing “a valid coded bitstream” that can be used “for compressing video sequences to bitrates around 1,5 Mbit/s.” MPEG-1, 12, 14, 22. In particular, the figure shows the “main functional blocks” of an encoder, including: a motion estimator block, a picture store/predictor block, a DCT block, a quantization block, a buffer, and a regulator block. MPEG-1, 14. MPEG-1 thus discloses or suggests *a video encoder for encoding a digital video sequence*, as recited in 12[pre].



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

MPEG-1, 14.

- (ii) **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;**

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

173. As I described in §IX.A.1 (MPEG-1 overview), 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*. The corresponding structure is a motion compensated predictor and combiner, as explained by the '148 Patent. '148Pat., 7:66-8:4 (motion compensated predictor),

8:5-9 (combiner).

174. As I described in §X.A.2.a(ii) (for 1[a]), MPEG-1 discloses *applying motion compensated prediction to blocks of pixels*, recited in 12[a]. Furthermore, Figure 2 (reproduced and annotated, below) illustrates that the video encoder includes a “Motion Estimator” block (purple) that generates motion vectors (light blue) provided to a “Picture store / Predictor” block (green). MPEG-1, 14.

175. With respect to the encoder, MPEG-1 describes:

The encoder must choose which picture type to use for each picture. Having defined the picture types, the encoder estimates motion vectors for each 16 by 16 macroblock in the picture. In P-Pictures one vector is needed for each non-intra macroblock and in B-Pictures one or two vectors are needed.

MPEG-1, 14.

176. In other words, referencing the encoder of Figure 2, MPEG-1 describes that “the encoder estimates motion vectors for each 16 by 16 macroblock in the picture.” MPEG-1, 14. A POSITA would therefore have understood that the “Picture store / Predictor,” which 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 described in the ’148 Patent.

177. MPEG-1 also discloses or suggests *provid[ing] corresponding blocks of prediction error values* for the reasons discussed in §X.A.2.a(ii) for 1[a] and the

substantially similar limitation discussed therein. MPEG-1's Figure 2 (reproduced and annotated, below) illustrates that the difference block (red) of the video encoder takes the difference (illustrated with minus "-" sign, light green) between input pictures (yellow) and the predictions formed by the "Picture store / Predictor" (orange). A POSITA would have known that this difference (blue) is *blocks of prediction error values*.

178. Further, the '148 Patent explains:

Each 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.

179. In other words, the '148 Patent explains that a combiner subtracts predicted pixel values from input pixel values. '148Pat., 8:5-9. The difference block (red) in Figure 2 similarly subtracts predicted pixel values (i.e., the result of motion compensated prediction) from input pixels (i.e., input pictures). The difference block therefore operates in the same manner as the combiner described by the '148 Patent. A POSITA therefore would have known that the difference block (red) in the MPEG-1 encoder that provides the blocks of prediction error values is a combiner that operates in the same manner as the combiner described in the '148 Patent.

182. MPEG-1 discloses or suggests *transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values*, as recited by 12[b], for the reasons I described in §X.A.2.a(iii) for 1[b] and the substantially similar limitation discussed therein. That is, 12[b] recites *transforming* (“applying a transform coding technique to” of 1[b]) *said blocks of prediction error values* (as also recited by 1[b]) *for providing sets of transform coefficient values* (“to produce sets of transform coefficient values” of 1[b]) *representative of said blocks of prediction error values* (as also recited by 1[b]), which MPEG-1 discloses or suggests.

183. Further, the encoder illustrated in Figure 2 of MPEG-1 (reproduced and annotated, below) includes a DCT block (red) that receives prediction error signals (blue) from the combiner. MPEG-1, 14-15.

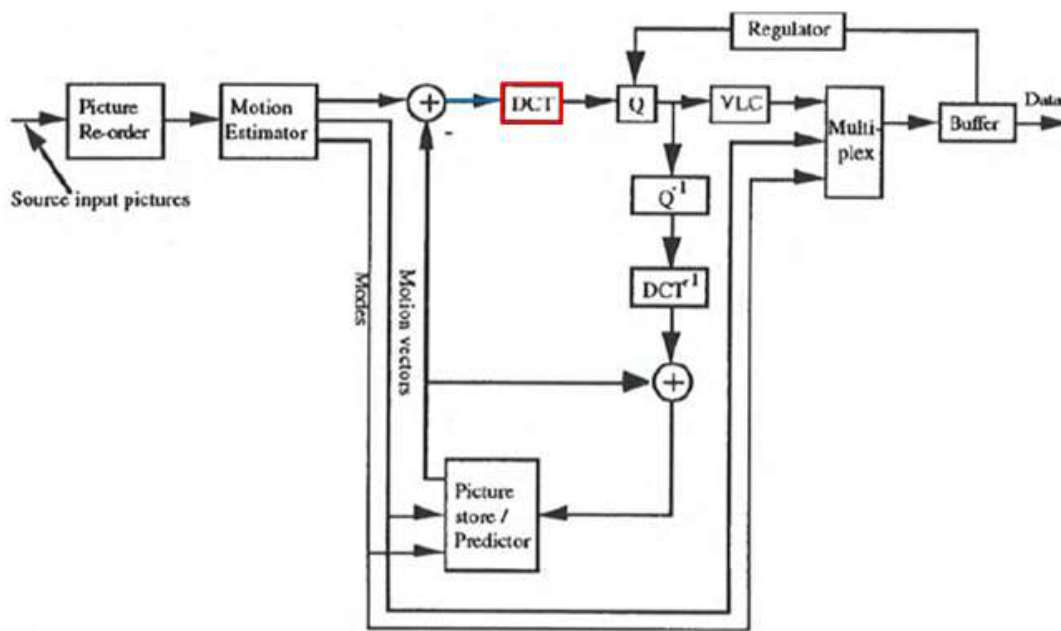
184. Referencing the encoder of Figure 2, MPEG-1 describes:

Firstly, for a given macroblock, the coding mode is chosen. It depends on the picture type, the effectiveness of motion compensated prediction in that local region, and the nature of the signal within the block. Secondly, depending on the coding mode, a motion compensated prediction of the contents of the block based on past and/or future reference pictures is formed. This prediction is subtracted from the actual data in the current macroblock to form an error signal. Thirdly, this error signal is separated into 8 by 8 blocks (4 luminance and 2 chrominance blocks in each macroblock) and a discrete cosine transform is performed on each block. Each resulting 8 by 8 block of DCT

coefficients is quantized and the two-dimensional block is scanned in a zig-zag order to convert it into a one-dimensional string of quantized OCT coefficients. Fourthly, the side-information for the macroblock (mode, motion vectors etc) and the quantized coefficient data are encoded. For maximum efficiency, a number of variable length code tables are defined for the different data elements. Run-length coding is used for the quantized coefficient data.

MPEG-1, 15.

185. In other words, MPEG-1 describes, referencing the encoder of Figure 1, 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. A POSITA would have understood that the illustrated DCT block, which receives the prediction error signals and generates the input to the quantizer (i.e., DCT coefficients), *provid[es] sets of transform coefficient values, recited in 12[b]*, from the received prediction error signals in the same manner as the transform block described in the ’148 Patent.



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

MPEG-1, 14.

- (iv) **12[c]: means for selecting a default level of quantization for quantizing said sets of transform coefficient values; and**

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

187. As I described in §IX.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, as explained by the '148 Patent. '148Pat., 6:16-22; 19:60-65.

188. MPEG-1 discloses or suggests *selecting a default level of quantization for quantizing said sets of transform coefficient values*, as recited by 12[c], for the

reasons I described in §X.A.2.a(iv) for 1[c] and the substantially similar limitation discussed therein. That is, 12[c] recites *selecting a default level of quantization* (“defining a default level of quantization” of 1[c]) *for quantizing* (“for use in encoding of the digital video sequence to quantize” of 1[c]) *said sets of transform coefficient values* (“the sets of transform coefficient values” of 1[c]), which MPEG-1 discloses or suggests.

189. Referencing the encoder of Figure 2, MPEG-1 describes:

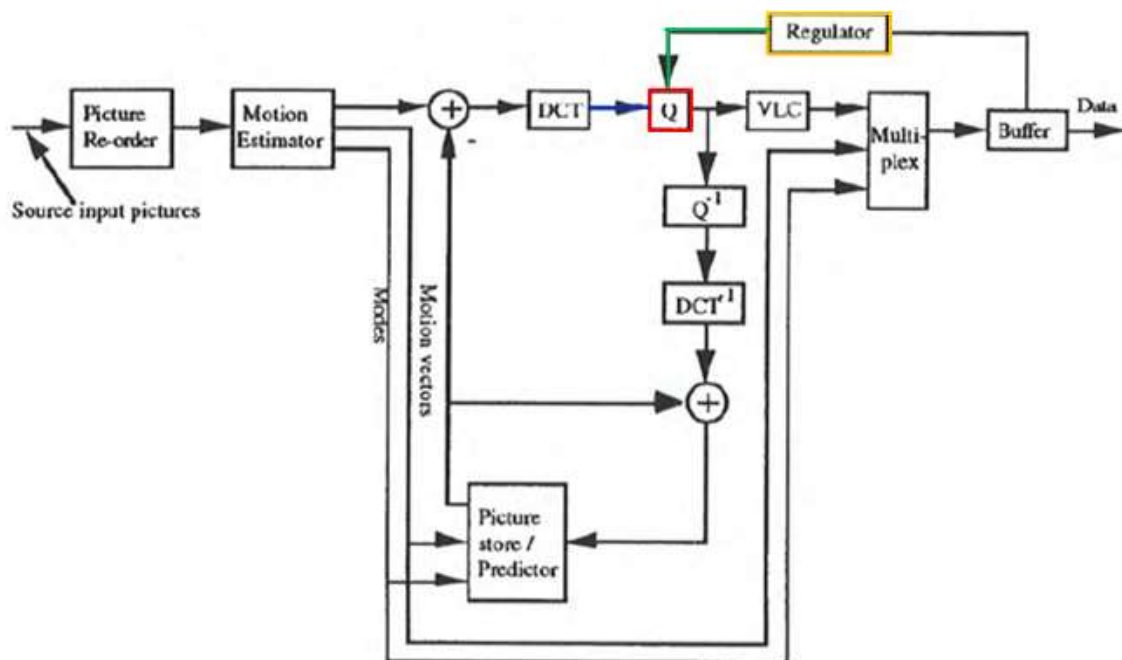
A consequence of using different picture types and variable length coding is that the overall data rate is variable. In applications that involve a fixed-rate channel, a FIFO buffer may be used to match the encoder output to the channel. The status of this buffer may be monitored to control the number of bits generated by the encoder. Controlling the quantization process is the most direct way of controlling the bitrate. This part of ISO/IEC 11172 specifies an abstract model of the buffering system (the Video Buffering Verifier) in order to constrain the maximum variability in the number of bits that are used for a given picture. This ensures that a bitstream can be decoded with a buffer of known size.

MPEG-1, 15.

190. In other words, MPEG-1 discloses in reference to the encoder of Figure 2, 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.

191. 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. 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. Therefore, A POSITA would have understood that the Regulator block performs the described “[c]ontrolling [of] the quantization process” to control the bitrate. Further, to control the quantization process, the Regulator must *select* the quantization used. 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*, as is recited in 12[c], the transform coefficient values by the quantization block. The Regulator block of the MPEG-1 operates in the same manner as the control manager described in the '148 Patent.



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

MPEG-1, 14.

- (v) **12[d]: means for providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence.**

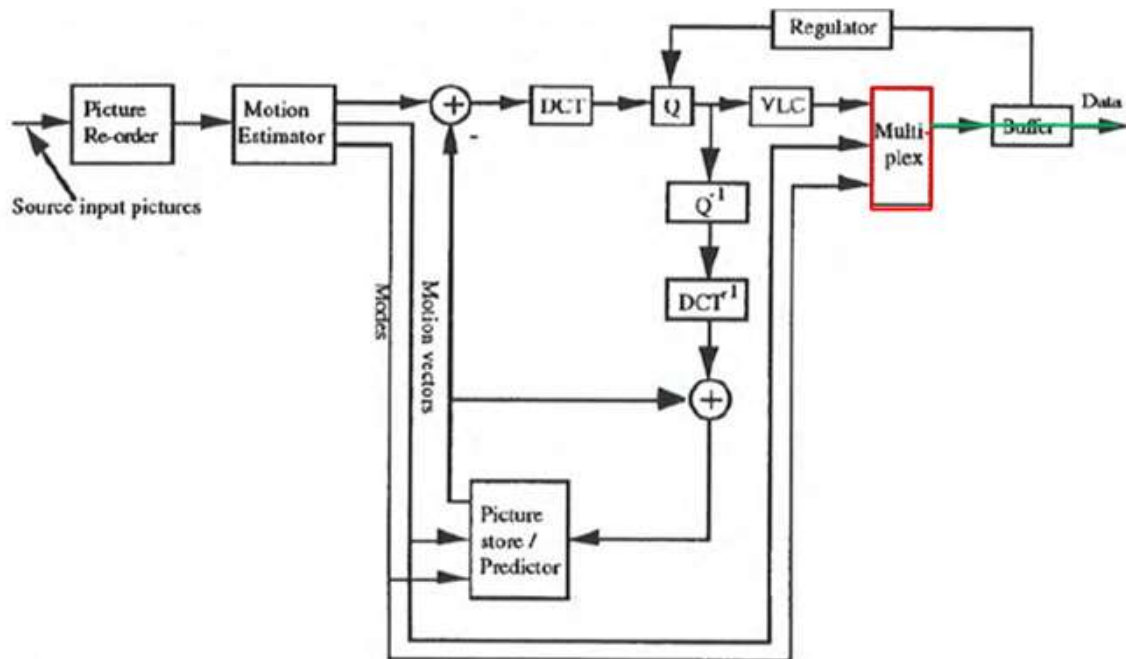
192. MPEG-1 discloses or suggests 12[d].

193. As I described in §IX.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, as explained by the '148 Patent. '148Pat., 6:38-47; 19:60-65.

194. MPEG-1 discloses or suggests *providing an indication of said default*

level of quantization in the encoded bit-stream representative of the digital video sequence, as recited by 12[d], for the reasons I described in §X.A.2.a(v) for 1[d] and the substantially similar limitation discussed therein. That is, 12[d] recites *providing an indication of said default level of quantization* (“providing an indication of the default level of quantization” of 1[d]) *in the encoded bit-stream representative of the digital video sequence* (“to a decoding process” of 1[d]), which MPEG-1 discloses or suggests.

195. The encoder illustrated in Figure 2 of MPEG-1 (reproduced and annotated, below) includes a multiplex block (red). MPEG-1, 14. As shown, 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). 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 described in the '148 Patent. MPEG-1, 14-15. The encoded bitstream includes the sequence level header that indicates *the default level of quantization*, as I described in §X.A.2.a(v) (for 1[d]).



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

MPEG-1, 14.

- b. **Claim 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.**

196. MPEG-1 discloses or suggests claim 13's additional limitation.

197. MPEG-1 discloses or 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 I described in §X.A.2.e (for claim 5), and because its encoder (i.e., video encoder) conforms to its encoding method.

c. Claim 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.

198. MPEG-1 discloses or suggests claim 14’s additional limitation.

199. MPEG-1 discloses or 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*, as recited by claim 14, for the reasons I described in §X.A.2.f for claim 6 and the substantially similar limitation discussed therein. That is, claim 14 recites *arranged to provide...in the encoded bit-stream* (“providing to a decoding process” of claim 6) *an indication of a difference between said level of quantization and the default level of quantization* (as also recited by claim 6), which MPEG-1 discloses or suggests.

200. Furthermore, a POSITA would have understood that MPEG-1’s disclosed encoder (i.e., *a video encoder*) performs MPEG-1’s encoding method that provides the indication in the encoded bit-stream sent to a decoder.

- d. Claim 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.**

201. MPEG-1 discloses or suggests claim 15's additional limitation.

202. MPEG-1 discloses or 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*, as recited by claim 15, for the reasons I described in §X.A.2.d for claim 4 and the substantially similar limitation discussed therein. That is, claim 15 recites *wherein the default level of quantization can be updated* (“wherein the default level of quantization is updated during the encoding of the digital video sequence” of claim 4) *and the video encoder is arranged to provide an indication of the updated default level of quantization in the encoded bit-stream* (“said method further comprising providing an indication of the updated default level of quantization to a decoding process” of claim 4), which MPEG-1 discloses or suggests.

203. Furthermore, a POSITA would have understood that MPEG-1's disclosed encoder (i.e., *a video encoder*) conforms to MPEG-1's encoding method that provides the indication *in the encoded bit-stream* sent to a decoder.

- e. **Claim 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.**

204. MPEG-1 discloses or suggests claim 16's additional limitation.

205. As discussed in §X.A.3.b for claim 13, 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.

Furthermore, 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), recited in claim 16.

- f. **Claim 17: A video encoder according to claim 12, provided in a multimedia terminal.**

206. MPEG-1 discloses or suggests claim 17's additional limitation.

207. 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. According to MPEG-1, the coded representation can be used for television, workstations, and personal computer displays. MPEG-1, 12.

208. With regard to a multimedia terminal, the '148 Patent explains:

FIG. 8 presents a terminal device comprising video encoding and decoding equipment which may be adapted to operate in accordance with the present invention. More precisely, the figure illustrates a multimedia terminal 80 implemented according to ITU-T recommendation H.324. The terminal can be regarded as a multimedia transceiver device. It includes elements that capture, encode and multiplex multimedia data streams for transmission via a communications network, as well as elements that receive, de-multiplex, decode and display received multimedia content.

'148Pat., 22:56-65.

This kind of multimedia terminal can be used in real-time applications such as conversational videotelephony, or non real-time applications such as the retrieval and/or streaming of video clips, for example from a multimedia content server in the Internet.

'148Pat., 23:1-5.

209. In other words, 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. The '148 Patent also 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.

210. 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). 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). In other words, MPEG-1 describes that the MPEG-1 standard may be utilized in systems (e.g., televisions, workstations, and personal computer displays) that perform the functions of a *multimedia terminal*. MPEG-1 therefore discloses or suggests that MPEG-1 can be implemented in systems including *a multimedia terminal*, recited in claim 17.

g. Claim 18: A video encoder according to claim 12, provided in a radio telecommunications device.

211. MPEG-1 discloses or suggests claim 18's additional limitation.

212. As I described in §X.A.3.f for (claim 17), MPEG-1 describes that the coded representation of MPEG-1 can be used for television, workstations, and personal computer displays. MPEG-1, 12. 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*, recited in claim 18.

- h. Claim 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.**

213. MPEG-1 discloses or suggests claim 19's additional limitation.

214. MPEG-1 discloses or 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*, as recited by claim 19, for the reasons I described in §X.A.2.g for claim 7 and the substantially similar limitation discussed therein. That is, claim 19 recites *wherein said level of quantization applied to said sets of transform coefficient values* (“wherein said level of quantization” of claim 7) *can be adjusted from one frame of the digital video sequence to another* (“is adjusted from one frame of the digital video sequence to another” of claim 7) *so as to apply an actual level of quantization to the sets of transform coefficients for a particular frame of the digital video sequence* (“such that an actual level of quantization applied to the sets of transform coefficients for a particular frame of the digital

video sequence” of claim 7) *and wherein said actual level of quantization is different from the default level of quantization* (“is different from the default level of quantization” of claim 7), which MPEG-1 discloses or suggests.

215. Moreover, a POSITA would have understood that MPEG-1’s encoder (i.e., *a video encoder*) illustrated in Figure 2, performs MPEG-1’s encoding method.

- i. **Claim 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.**

216. MPEG-1 discloses or suggests claim 20’s additional limitation.

217. MPEG-1 discloses or suggests *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*, as recited by claim 20, for the reasons I described in §X.A.2.i for claim 9 and the substantially similar limitations discussed therein. That is, claim 20 recites *arranged to provide an indication...in the encoded bit-stream representative of the digital video sequence* (“further comprising providing to a decoding process an indication” of claim 9) *of said difference with respect to the default level of quantization* (“of a difference between said actual level of quantization and the default level of quantization” of claim 9), which MPEG-1 discloses or suggests.

218. Moreover, a POSITA would have understood that MPEG-1’s encoder

(i.e., *a video encoder*) illustrated in Figure 2, performs MPEG-1's encoding method.

- j. **Claim 21: 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 to adjust the level of quantization applied to said sets of transform coefficient values from one segment of a frame to another 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, 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.**

219. MPEG-1 discloses or suggests claim 21's additional limitation.

220. 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 §X.A.2.j and §X.A.2.k, respectively). 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 or suggests claim 21's additional limitation for the reasons

discussed in §X.A.2.j and §X.A.2.k. Furthermore, a POSITA would have understood that MPEG-1's encoder (i.e., *a video encoder*) performs MPEG-1's encoding method.

Claim 21		Corresponding Elements from Claims 10 and 11
Reference	Claim Limitation	
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	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; (Claim 10, §X.A.2.j)
21[b]	to adjust the level of quantization applied to said sets of transform coefficient values from one segment of a frame to another	wherein the level of quantization applied to said sets of transform coefficient values is adjusted from one segment of a frame to another; (Claim 10, §X.A.2.j)
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,	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 10, §X.A.2.j)
21[d]	said video encoder further arranged to represent the actual level of quantization for use in the particular segment as a	providing to a decoding process an indication of a difference between said level of quantization and the default level of

	difference with respect to the default level of quantization.	quantization. (Claim 11, §X.A.2.k)
--	---	------------------------------------

- k. Claim 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.**

221. MPEG-1 discloses or suggests claim 22's additional limitation.

222. MPEG-1 discloses or suggests *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*, as recited by claim 22, for the reasons I described in §X.A.2.k for claim 11, and the substantially similar limitation discussed therein. That is, claim 22 recites *further arranged to provide an indication...in the encoded bit-stream representative of the digital video sequence* (“further comprising providing to a decoding process an indication” of claim 11) *of said difference with respect to the default level of quantization* (“of a difference between said level of quantization and the default level of quantization” of claim 11), which MPEG-1 discloses or suggests.

223. Moreover, a POSITA would have understood that MPEG-1's encoder (i.e., *a video encoder*) illustrated in Figure 2, performs MPEG-1's encoding method.

4. Claim 23

- a. **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:**

224. To the extent the preamble is limiting, MPEG-1 discloses or suggests 23[pre] for substantially the same reasons as 12[pre]. §X.A.3.a(i).

- b. **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;**

225. MPEG-1 discloses or suggests 23[a].

226. As I described in §IX.A.1, the function of the *module for encoding* is *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*, and the structure is the motion compensated predictor and combiner. The function and structure are identical to the function of structure of 12[a], and therefore MPEG-1 discloses or suggests this limitation for the same reasons discussed in X.A.3.a(ii).

- c. **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**

227. MPEG-1 discloses or suggests 23[b].

228. As I described in §IX.A.2, the function of the *module for transforming* is *providing sets of transform coefficient values representative of said blocks of prediction error values*, and the structure is the transform block. The function and structure are identical to the function of structure of 12[b], and therefore MPEG-1 discloses or suggests this limitation for the same reasons discussed in §X.A.3.a(iii).

- d. **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.**

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

230. As I described in §IX.A.3, the function of the *module for selecting* includes *selecting a default level of quantization for quantizing said sets of transform coefficient values*, and the structure is the control manager. The noted function and structure are identical to the function of structure of 12[c], and therefore MPEG-1 discloses or suggests the noted limitation for the same reasons as discussed in §X.A.3.a(iv).

231. Additionally, 23[c] recites *wherein the default level is specific to the video coding applications*. The sole disclosure in the '148 patent of this 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 particular, according to a second possible embodiment of the invention, a sequence QP is not transmitted but an application specific constant is used as sequence QP instead.

’148Pat, 25:25-27.

232. In other words, according to the ’148 Patent, a default level of quantization is consistent for video sequences processed by a video application.

233. As I explained in §X.A.2.a(iv) (for 1[c]), 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. When load_non_intra_quantizer_matrix is set to 0, values for the non-intra quantizer matrix are not transmitted. MPEG-1, 42. Accordingly, when an MPEG-1 video sequence does not specify any non-intra quantizer matrix values (i.e., the load_non_intra_quantizer_matrix is not set to 1 during the sequence and 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., the MPEG-1 video coding standard).

234. Furthermore, the default level (of the flat default quantization matrix) may be used by any video coding application that implements MPEG-1 encoding. 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. '148Pat, 25:25-27.

B. Ground 2: Obviousness over H.263

1. H.263

235. ITU-T Recommendation H.263 is titled “Video coding for low bit rate communication.” The revision of H.263 I reference herein (otherwise known as H.263+, H.263 (1998), or the 1998 version of H.263) was ratified by the ITU-T in February 1998, and I understand was made available to the public no later than May 11, 2000. H.263, i; Munford, 3-9. I understand that H.263 is therefore prior art to the '148 patent.

236. H.263 “specifies a coded representation that can be used for compressing the moving picture component of audio-visual services at low bit rates.” H.263, 1. That is, H.263 is a video compression standard.

237. H.263 defines a “source format” for the pictures (of a video sequence) to be encoded, where “[p]ictures are coded as luminance and two colour difference components (Y , C_B and C_R).” H.263, 7. A POSITA would have understood that the luminance component represents brightness, and the colour difference components (also referred to as chrominance) represents color content. '148Pat., 1:34-38; H.263, 7-8.

238. In H.263, “[t]he video syntax is arranged in a hierarchical structure with four primary layers.” H.263, 13. From top to bottom, the layers are: (1) picture; (2) Group of Blocks (GOB), or slice, or video picture segment; (3) macroblock; and (4) block. H.263, 13. Each of the blocks represents luminance or chrominance. H.263, 11.

239. In H.263, “[t]he primary form of prediction is inter-picture and may be augmented by motion compensation.” H.263, 11. When inter-picture prediction (also referred to as INTER coding) is used for a picture (referred to as a P-picture) and/or macroblocks therein, the prediction is based on a “temporally previous reference picture.” H.263, 11. In contrast, when INTRA coding (also called an I-picture) is used, no temporal prediction is applied. H.263, 11. When the inter-picture prediction (e.g., INTER coding) is augmented by motion compensation, by default “[t]he decoder will accept one [motion] vector per macroblock.”⁵ H.263, 11-12.

240. One of the “main elements” of H.263’s video source coding (i.e., encoding) algorithm is “block transformation,” in which a transform is applied to

⁵ In sections of the H.263 standard that pertain to motion compensation, a POSITA would understand that the terms *vector* and *macroblock vector* generally mean *motion vector* unless otherwise specified.

the blocks forming each macroblock to generate transform coefficients. H.263, 9, 11, Fig. 3. H.263 additionally describes how a decoder reconstructs each luminance block and chrominance block based on the result of applying an inverse transform (i.e., an inverse DCT) to encoded blocks. H.263, 47. H.263 explains that “[f]or INTRA blocks, the reconstruction is equal to the result of the inverse transformation.” H.263, 47. In contrast, H.263 explains that “[f]or INTER blocks, the reconstruction is formed by summing the prediction and the result of the inverse transformation.” H.263, 47-48. In further describing the reconstruction operation, H.263 alternatively refers to the result of the inverse transformation for an INTER block as “reconstructed prediction error.” H.263, 48. In other words, H.263 describes that the inverse transform of an encoded INTRA block yields a luminance or chrominance value, and the inverse transform of an encoded INTER block yields a prediction error.

241. Though H.263 describes the values yielded by applying an inverse transform to encoded INTRA and INTER blocks (luminance or chrominance values, and prediction error, respectively) under the heading of “Decoding process” (H.263, 44), it would have been obvious to a POSITA that an H.263 encoder transforms (during encoding) the same values (luminance, chrominance, or prediction error) as specified under H.263’s “Decoding process.” That is, if an H.263 decoder will use the result of an inverse transform as a prediction error, then an H.263 encoder would have applied its transform to the prediction error. H.263

indicates the same, when it describes how “[t]he video coder provides a self-contained digital bitstream...The video decoder performs the reverse process.”

H.263, 2. In other words, H.263’s description of the values that result by an inverse transform during a decoding process also illustrates what values are first transformed during an encoding process. H.263 therefore describes transforming luminance and chrominance values for INTRA blocks, and transforming prediction error for INTER blocks. As I described above in §VII (Technology Overview), the prediction error represents the difference between a frame and the decoded version of the frame (i.e., the difference between the values of the luminance and chrominance blocks, and their respective predictions).

242. H.263 further describes that the transform coefficients (generated by applying a transform to luminance values, chrominance values, or prediction error, depending on the type of block) are quantized, and the resulting quantized transform coefficients are further encoded (e.g., entropy coded) before being transmitted (e.g., to a decoder). H.263, 9, 11-13, 46, Fig. 3. Quantization is performed using a quantizer value, which H.263 refers to as “QUANT.” H.263, 12-13, 30. As I discuss in greater detail below, QUANT is set by quantization information specified 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.

2. Claim 1 and Dependents

a. Claim 1

- (i) **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:**

243. To the extent the preamble is limiting, H.263 discloses or suggests 1[pre].

244. As I described in §X.B.1 (H.263 overview), H.263 “specifies a coded representation that can be used for compressing the moving picture component of audio-visual services at low bit rates.” H.263, 1. H.263 therefore discloses or suggests *a method of encoding a digital video sequence*. And H.263 illustrates the environment in which the coded representation can be used by describing a “video codec” that includes a video coder and video decoder (also referred to as a source coder and source decoder, respectively), where the “video coder provides a self-contained digital bitstream.” H.263, 2, Fig. 1. H.263 therefore discloses or suggests *for use in a video coding application to produce an encoded video bit-stream representative of the digital video sequence*.

245. The H.263 “source coder operates on non-interlaced pictures occurring at a Picture Clock Frequency (PCF) of 30 000/1001 (approximately 29.97) times

per second.” H.263, 7. Each non-interlaced picture is a *frame*, and therefore H.263 discloses or suggests *the digital video sequence comprising a number of frames*.

246. As I described in §X.B.1 (H.263 overview), the pictures (*frames*) on which H.263 operates are coded as luminance (Y) and color difference (CB and CR) components. H.263, 7. Further, H.263 explains that “[t]he picture size is measured as the picture width times height for the luminance (Y) component, measured in pixels.” H.263, 6. H.263 therefore discloses or suggests *each frame of said sequence comprising an array of pixels*.

247. Each picture is divided into GOBs, and each GOB is divided into macroblocks. H.263, 9, 11. Further:

[E]ach 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 . Further, a macroblock consists of four luminance blocks and the two spatially corresponding colour difference blocks as shown in Figure 5. Each luminance or chrominance block thus relates to 8 pixels by 8 lines of Y, C_B or C_R .

H.263, 11.

248. That is, each macroblock of a picture (*frame*) consists of luminance blocks and chrominance blocks (each a *block*), where each chrominance block and luminance block relates to 8 pixels by 8 lines (*each block comprising a certain number of said pixels*). H.263 therefore discloses or suggests *each frame of said sequence ... divided into a plurality of blocks, each block comprising a certain number of said pixels*.

- (ii) **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;**

249. H.263 discloses or suggests 1[a].

250. As I described in §X.B.2.a(i) (for 1[pre]), in H.263 each picture (*frame*) consists of macroblocks, each of which consists of four luminance blocks and two chrominance blocks (each a *block of pixels*).

251. As I described in §X.B.1 (H.263 overview), in H.263 “[t]he primary form of prediction is inter-picture and may be augmented by motion compensation.” H.263, 11. When inter-picture prediction (also referred to as INTER coding) is used for a picture (referred to as a P-picture) and/or macroblocks therein, the prediction is based on a “temporally previous reference picture.” H.263, 11. H.263 describes how “[motion vector data] is included for all INTER macroblocks ... and consists of a variable length codeword for the horizontal component followed by a variable length codeword for the vertical component.” H.263, 38. H.263 further explains that “[a] positive value of the horizontal or vertical component of the motion vector signifies that the prediction is formed from pixels in the referenced picture which are spatially to the right or below the pixels being predicted.” H.263, 12. In other words, H.263 describes that each INTER macroblock is encoded with a motion vector, which indicates the spatial relationship between the pixels being predicted (corresponding to the macroblock)

and the pixels (in the temporally previous reference picture) from which the prediction is formed.

252. In describing how motion vectors are used during motion compensation (H.263, 44), H.263 explains that:

The motion vector is used for all pixels in all four luminance blocks in the macroblock. Motion vectors for both chrominance blocks are derived by dividing the component values of the macroblock vector by two, due to the lower chrominance format.

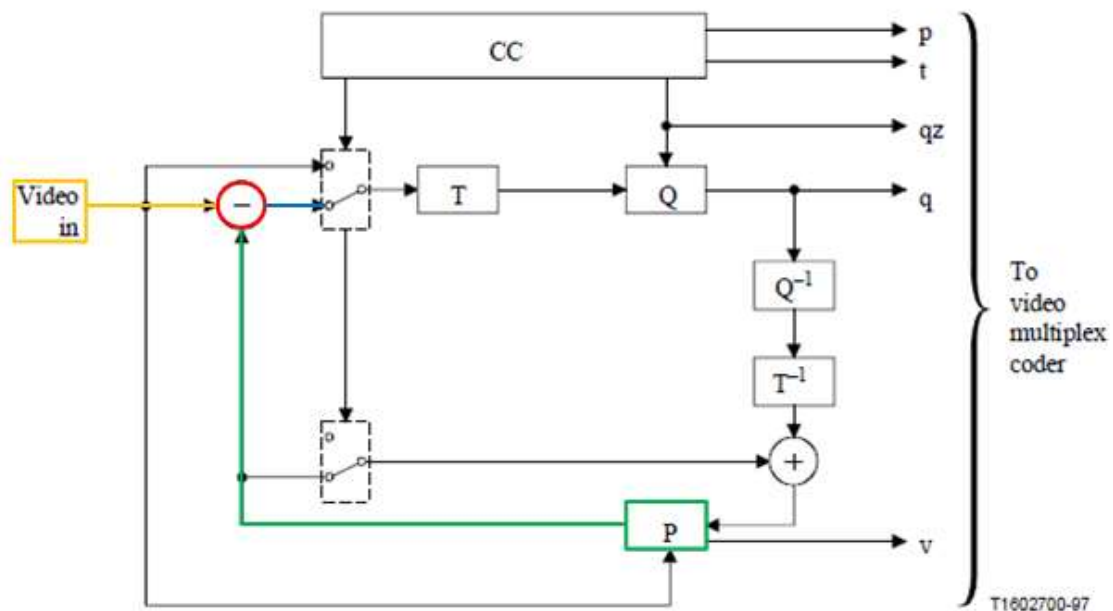
H.263, 45.

253. In other words, a macroblock's motion vector is used (*applying motion compensated prediction*) for the chrominance blocks and luminance blocks therein (each a *block of pixels*). H.263 therefore discloses or suggests *encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels*.

254. 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 ’148 Patent explains that *prediction error* “represents the difference between a decoded version of the INTER-coded frame and the image content of the frame to be coded” (’148Pat., 3:33-36), and H.263’s “remaining signal” is a *prediction error*. This is consistent with how a POSITA would have understood both terms.

255. Moreover, and as I described in §X.B.1 (H.263 overview), H.263 describes that during a decoding process, each of the chrominance blocks and luminance blocks of INTER macroblocks are reconstructed “by summing the prediction and the result of the inverse transformation. The summation is performed on a pixel basis” H.263, 44, 47-48. H.263 also refers to the value being summed with the prediction (i.e., the result of the inverse transformation) as “reconstructed prediction error.” H.263, 48. That is, the “result of the inverse transformation” of each block of an INTER macroblock is *prediction error*, and thus H.263 reconstructs *prediction error values* when reconstructing chrominance blocks and luminance blocks (*blocks of pixels*) of an INTER macroblock.

256. H.263 further illustrates this step in the illustration of its encoder. The encoder’s “main elements are prediction, block transformation and quantization.” H.263, 9. H.263’s Figure 3 (reproduced and annotated below) illustrates the “generalized form” (H.263, 9) of its encoder, in which the encoder generates (in red) the difference between the video input (in orange) and the result of motion compensated prediction (in green). This difference (in blue) corresponds to the *prediction error*.



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

H.263, Fig. 3 (annotated)

257. H.263 thus discloses or suggests *producing corresponding blocks of prediction error values* based on the applied *motion compensated prediction*.

(iii) **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;**

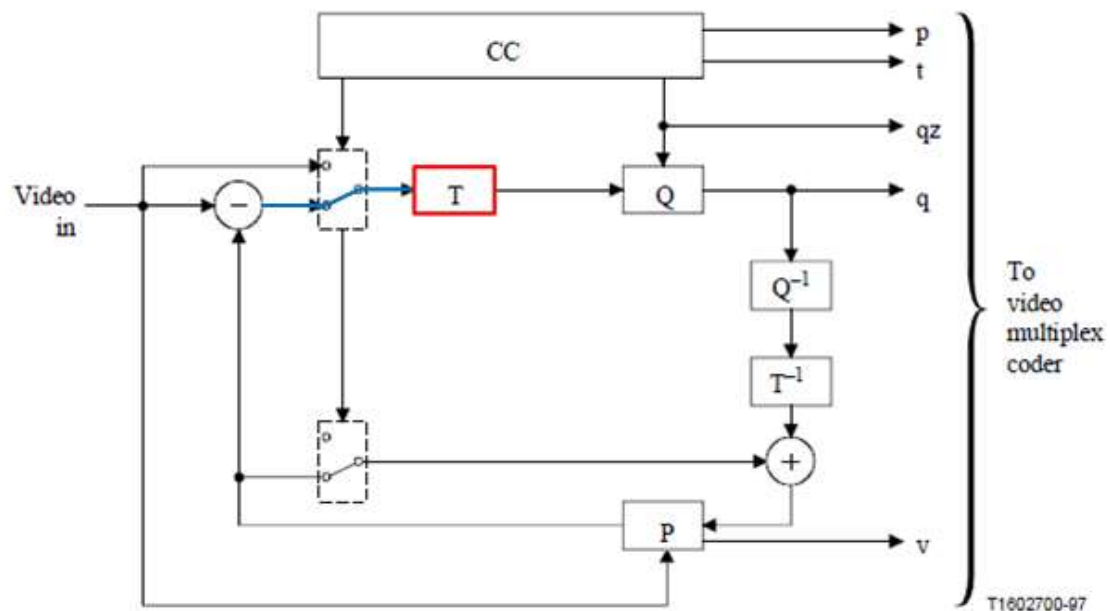
258. H.263 discloses or suggests 1[b].

259. H.263 describes that its encoding algorithm utilizes “[a] hybrid of

inter-picture prediction to utilize temporal redundancy and transform coding of the remaining signal to reduce spatial redundancy.” H.263, 2. H.263 further describes how when decoding, an inverse transform such as inverse DCT is used. H.263, 47. H.263 therefore discloses or suggests *applying a transform coding technique*. Further, for the reasons I described in §X.B.1, the *transform coding technique* may be the DCT.

260. As I described in §X.B.2.a(ii) (for 1[a]), the “remaining signal” to which the *transform coding technique* is applied is a *prediction error*. And as I additionally described in §X.B.2.a(ii), H.263 describes reconstructing *prediction error values* for each chrominance block and luminance block (*to said blocks*) of a coded INTER macroblock (e.g., at decode), and therefore produces *prediction error values* for each chrominance block and luminance block (*to said blocks*) of the INTER macroblock.

261. H.263’s Figure 3 (reproduced and annotated below), illustrates the same, illustrating that INTER coding applies a transform (in red) to the prediction error values (in blue).



- 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

H.263, Fig. 3 (annotated).

262. H.263 further describes the transform coding as a “block transformation,” in which “[t]ransmitted blocks are transformed.” H.263, 9, 11. In other words, the *transform coding technique* (such as DCT) is applied to each of the blocks of a macroblock, where—in the case of an INTER macroblock—each block is the produced *prediction error values*. H.263 therefore discloses or suggests applying the transform coding technique (e.g., DCT) *to produce sets of transform coefficient values representative of said blocks of prediction error values*.

(iv) 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

263. H.263 discloses or suggests 1[c].

264. I understand that in a prior co-pending litigation, Patent Owner has pointed to H.264 parameters “pic_init_qp_minus26,” “slice_qp_delta,” and “mb_qp_delta” as one or more examples of *defining a default level of quantization for use in encoding of the digital video sequence*, as claimed. Nokia Complaint, 415-16, 424-30. As I illustrate in greater detail below, these H.264 parameters set quantization information at the picture, slice, and macroblock layer, respectively.

265. The H.264 parameter pic_init_qp_minus26 is part of the “Picture parameter set,” and specifies the “initial” quantization information for slices in a picture. H.264, 32, 57. That is, the parameter is specified at the picture layer.

266. The H.264 parameter slice_qp_delta is part of the “Slice header syntax,” and specifies the “initial” quantization information for macroblocks in a slice. H.264, 36, 64. That is, the parameter is specified at the slice layer.

267. The H.264 parameter mb_qp_delta is part of the “Macroblock layer syntax,” and changes the quantization used for a macroblock. H.264, 41, 76. That is, the parameter is specified at the macroblock layer.

268. Just as in H.263, H.264’s layers are in a “hierarchical relationship,” in which “[h]igher layers contain lower layers.” H.264, 4. H.264’s coding layers (from

higher to lower) are “the coded video sequence, picture, slice, and macroblock layers.” H.264, 4. In other words, the highest-layer quantization parameter of H.264 that Patent Owner points to (from `pic_init_qp_minus26`, `slice_qp_delta`, and `mb_qp_delta`) is `pic_init_qp_minus26`, which is found at the picture layer.

269. Moreover, I have reviewed H.264, and find no evidence of a quantization parameter specified at a layer higher than the picture layer (e.g., the sequence layer). H.264 defines a “Sequence parameter set,” and none of the recited parameters specify quantization information. H.264, 31, 53-56. In other words, the picture layer is the highest layer of H.264 at which quantization information is specified.

270. Therefore, to the extent that Patent Owner argues that a picture-layer quantization parameter (e.g., `pic_init_qp_minus26` of H.264) is *a default level of quantization for use in encoding of the digital video sequence*, H.263’s quantization parameter PQUANT discloses or suggests the limitation under the same interpretation, as I describe in greater detail below.

271. In describing the operations of its “[v]ideo source coding algorithm” (H.263, 9), H.263 explains that “[t]ransmitted blocks are transformed and resulting coefficients are quantized and entropy coded.” H.263, 11. As I described in §X.B.1 (H.263 overview), quantization is performed using the quantizer QUANT.

272. H.263 further illustrates how QUANT is used when performing inverse quantization (e.g., during a decoding process), to reconstruct the coefficients that

were quantized during encoding, with references to the following formulas:

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

H.263, 46.

273. In the formulas above, QUANT refers to the quantizer used, LEVEL refers to the coded (i.e., quantized) transform coefficient, and |REC| refers to the unsigned reconstructed transform coefficient. H.263, 30, 41, 46. Additionally, in describing generally the quantization performed during encoding, H.263 references the inverse quantization formulas above, used during a decoding process, “[f]or the exact formulas.” H.263, 12-13. And a POSITA would have understood that quantization performed during encoding, and inverse quantization performed during decoding, would utilize related (and inverse) formulas. For this additional reason, H.263 discloses or suggests using the quantizer QUANT during encoding.

274. H.263 further describes that “the number of quantizers is 1 for the first coefficient of INTRA blocks and 31 for all other coefficients.” H.263, 12. In other words there are 31 possible quantizers that can be used when encoding INTER blocks, where the one used is indicated by QUANT. Since H.263 describes quantization during encoding, and further describes 31 possible quantizers, H.263 discloses or suggests *defining* a quantizer (QUANT) during encoding.

275. As I described in §X.B.1 (H.263 overview), H.263 describes a hierarchical structure with the following layers: (1) picture, (2) GOB, (3)

macroblock, and (4) block. Quantizer information, referred to as PQUANT, is provided in the picture layer. H.263, 22, 30, Fig. 6, Fig. 7. H.263 describes that 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.

276. PQUANT therefore indicates which of the 31 possible quantizers to use—by setting QUANT to a value from 1 to 31—for a picture.

277. Describing the picture layer, H.263 explains that “[d]ata for each picture consists of a picture header followed by data for [GOBs] or for slices, eventually followed by an optional end-of-sequence code and stuffing bits. The structure is shown in Figure 7[.]” H.263, 22. H.263’s Figure 7, reproduced below, shows that PQUANT is included in the picture header:

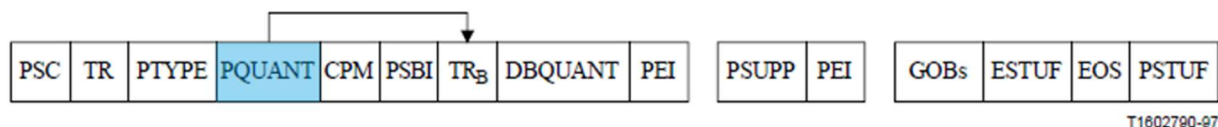


Figure 7/H.263 – Structure of picture layer (without optional PLUSPTYPE-related fields)

H.263, Fig. 7.

278. That is, each picture includes a picture header, which includes PQUANT. Therefore, every picture includes PQUANT.

279. As I described in §X.B.1 (H.263 overview), quantization information in H.263 may be specified at the GOB layer by GQUANT and/or at the macroblock layer by DQUANT. But in contrast to PQUANT, which is provided with each picture, neither GQUANT nor DQUANT are required to be included in an encoded bitstream. That is, as I describe in more detail below, H.263 does not require that additional quantization information be specified at the GOB layer or macroblock layer.

280. Describing the GOB layer, H.263 explains that “[d]ata for each [GOB] consists of a GOB header followed by data for macroblocks. The structure is shown in Figure 9.” H.263, 32. H.263’s Figure 9, reproduced below, shows that GQUANT is part of the GOB header:



Figure 9/H.263 – Structure of GOB layer

H.263, Fig. 9.

281. H.263 further explains that “[f]or the first GOB in each picture (with number 0), no GOB header shall be transmitted. For all other GOBs, the GOB header may be empty, depending on the encoder strategy.” H.263, 32. Since the first GOB of a picture does not transmit a GOB header, and all other GOBs within the picture may have empty GOB headers, GQUANT may therefore not be provided with any GOBs of a picture (e.g., when all GOBs of a picture, after the first GOB, have empty GOB headers).

282. Describing the macroblock layer, H.263 explains that DQUANT is “present when indicated by MCBPC.” H.263, 33. MCBPC, which stands for “Macroblock type & Coded Block Pattern for Chrominance,” “is a variable length codeword giving information about the macroblock type and the coded block pattern for chrominance.” H.263, 34. In other words, DQUANT is only included for certain macroblock types, as indicated by MCBPC. This is further illustrated in H.263’s Table 9, reproduced below, which shows that DQUANT is only present in certain macroblocks, as indicated by the macroblock type:

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.

H.263, Table 9.

283. That is, PQUANT is a quantization parameter that defines the quantizer (QUANT) for a picture, which may be optionally updated by GQUANT or DQUANT parameters specified for GOBs or macroblocks, respectively, within the picture. In other words, in H.263 every picture includes PQUANT to specify the

quantizer (i.e., sets QUANT) to be used for the picture. Further, as I described immediately above, no GOBs or macroblocks within the picture necessarily include GQUANT or DQUANT, respectively, to subsequently update QUANT for the picture. The quantizer set by PQUANT may therefore be used for quantizing the entirety of a picture. H.263's PQUANT therefore discloses or suggests *defining a default level of quantization*. A POSITA would find it obvious to set PQUANT at the same value for a series of pictures, which would essentially be a default level of quantization, rendering obvious the use of a sequence-level parameter to the extent required by the Board's interpretation of the claim language for claim 1.

284. As I described above in my discussion of this claim limitation, I understand that Patent Owner has pointed to H.264 parameters that set quantization information at the picture, slice, and macroblock layer as one or more examples of *defining a default level of quantization for use in encoding of the digital video sequence*. To the extent Patent Owner 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.

285. H.263 additionally explains that “[w]ithin a macroblock the same quantizer is used for all coefficients except the first one of INTRA blocks.” H.263, 12-13. That is, the same quantizer (QUANT) is used for all of the coefficients associated with the luminance and chrominance blocks that make up an INTER

macroblock. H.263 further illustrates the same by way of describing its decoding process, in which the reconstruction “for each luminance and chrominance block” of a macroblock is based on the results of inverse quantization using the quantizer QUANT. H.263, 46-47. H.263 therefore discloses or suggests a default level of quantization (e.g., QUANT as specified by PQUANT) is used to *quantize the sets of transform coefficient values* (e.g., the coefficients associated with the luminance blocks and chrominance blocks of an INTER macroblock).

(v) 1[d]: providing an indication of the default level of quantization to a decoding process.

286. H.263 discloses or suggests 1[d].

287. As I described in §X.B.2.a(iv) (for 1[c]), H.263’s PQUANT discloses a *default level of quantization*. And as I additionally described in §X.B.2.a(iv), PQUANT is included in the header for each picture. H.263 therefore discloses or suggests *providing an indication of the default level of quantization*.

288. As I further described in §X.B.2.a(iv), H.263 also describes performing inverse quantization during a decoding process. In particular, inverse quantization is performed using quantizer QUANT. H.263, 44, 46. And as I described in §X.B.2.a(iv) (for 1[c]), PQUANT indicates the quantizer QUANT to be used (i.e., PQUANT sets QUANT) for encoding and decoding. H.263 thus discloses or suggests *providing PQUANT to a decoding process*.

- b. Claim 2: An encoding method according to claim 1, wherein said indication of the default level of quantization is provided in the encoded bit-stream representative of the digital video sequence.**

289. H.263 discloses or suggests claim 2's additional limitation.

290. As I described in §X.B.2.a(v) (for 1[d]), H.263 discloses that PQUANT (*said indication of the default level of quantization*) is included in each picture's header. And as I described in §X.B.2.a(i) (for 1[pre]), H.263 describes that its video coder produces "a self-contained digital bitstream."

291. H.263 further shows the same in its syntax diagram that illustrates the elements of picture layer data. Figure 6 (part 1 of 7) of H.263, reproduced and annotated below, shows that the picture layer data includes PQUANT (in red) in a "video bitstream." H.263 therefore discloses or suggests that PQUANT is *provided in the encoded bit-stream representative of the digital video sequence*.

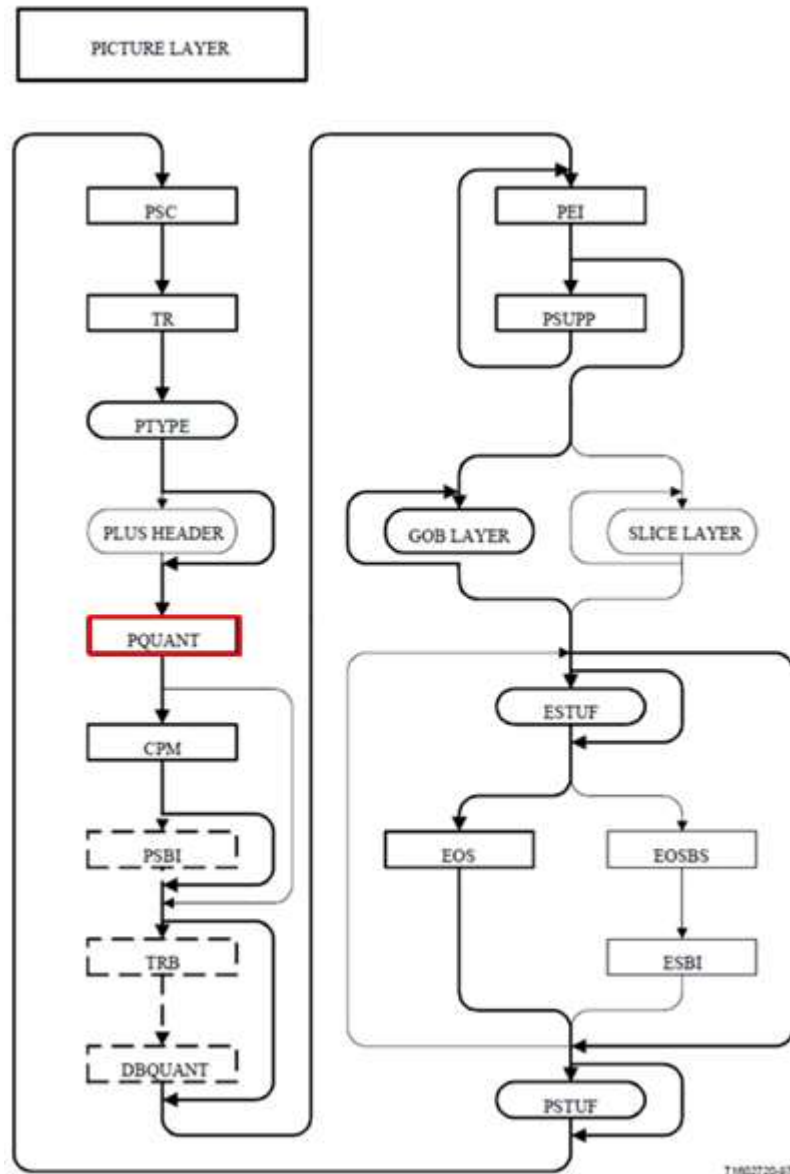


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

H.263, Fig. 6 (part 1 of 7) (annotated).

- c. **Claim 3: An encoding method according to claim 2, wherein the encoded bit-stream is transmitted from a video encoding device to a corresponding video decoding device.**

292. H.263 discloses or suggests claim 3’s additional limitation.

293. H.263 describes how its “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.

294. As reflected in Figure 1 of H.263, reproduced and annotated below, which illustrates the H.263 video coding system, a video coder (“Source coder,” red) (*video encoding device*) passes data to a “Transmission buffer” (blue) that is output as a “Coded bitstream” (purple). The figure further shows that the “Coded bitstream” (purple) is received by a “Receiving buffer” (green) and is passed to a “Source decoder” (light blue) (*video decoding device*). Thus the “Coded bitstream” (*encoded bit-stream*) is transmitted from an encoder to a decoder.

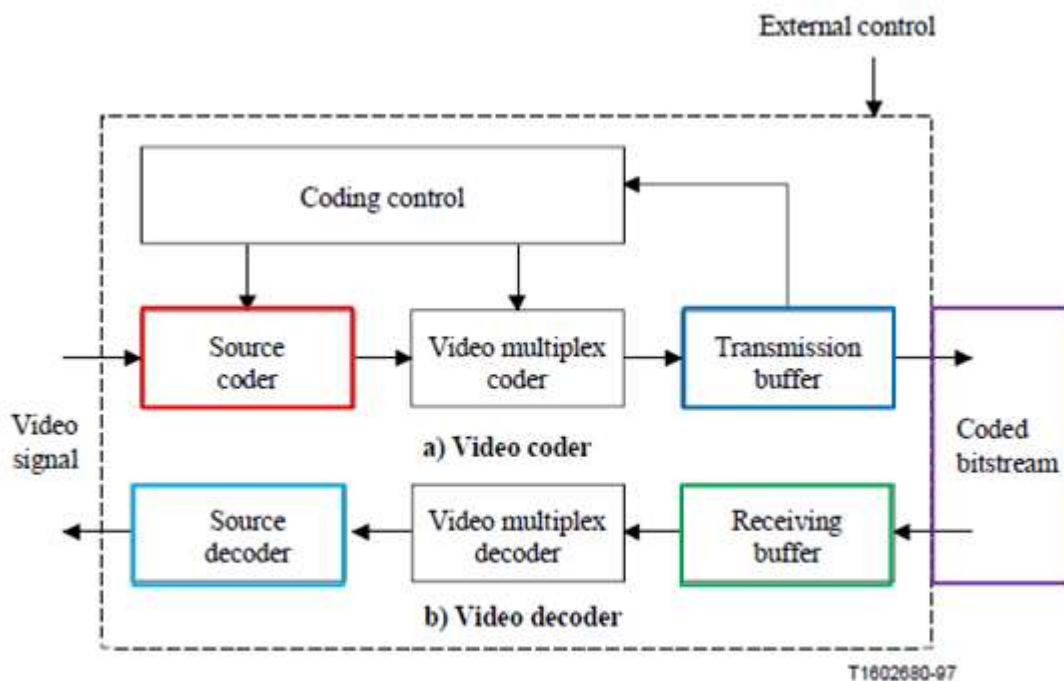


Figure 1/H.263 – Outline block diagram of the video codec

H.263, Fig. 1 (annotated).

295. H.263 therefore discloses or suggests that the coded bitstream (*encoded bit-stream*) is *transmitted from a video encoding device to a corresponding video decoding device*.

d. Claim 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.

296. H.263 discloses or suggests claim 4's additional limitation.

297. As I described in §X.B.2.a(iv) and §X.B.2.a(v) (for 1[c] and 1[d], respectively), H.263 discloses that PQUANT (*default level of quantization*) is included in the picture header. H.263 additionally describes that “[d]ata for each picture consists of a picture header followed by data for [GOBs] or for slices, eventually followed by an optional end-of-sequence code and stuffing bits.” H.263, 22. In other words, and as I additionally described in §X.B.2.a(iv), PQUANT is provided with each picture.

298. Further, H.263 “operates on non-interlaced pictures occurring at a Picture Clock Frequency (PCF) of 30 000/1001 (approximately 29.97) times per second.” H.263, 7. Therefore, as illustrated in H.263’s Figure 6 (part 1 of 7), reproduced and annotated below, each picture of a video sequence occurs at a PCF (green), contains picture layer data including PQUANT (red). PQUANT is

therefore provided with each picture of a video sequence (e.g., approximately 29.97 times per second), and specifies the *default level of quantization* for that picture (as I described in §X.B.2.a(iv), for 1[c]).

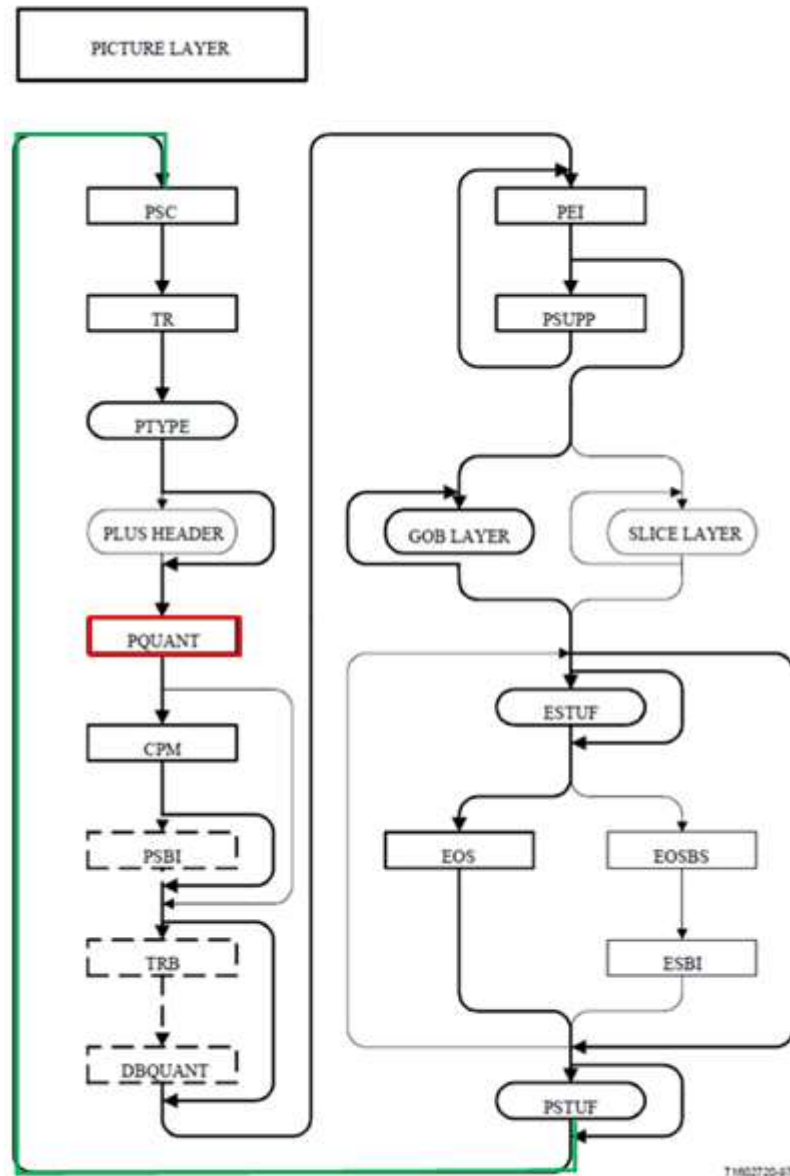


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

H.263, Fig. 6 (part 1 of 7) (annotated).

299. H.263 explains that PQUANT “indicates the quantizer QUANT to be used for the picture.” H.263, 30. A POSITA would have understood that PQUANT

could be set differently for different pictures, thereby indicating different quantizers to be used for different pictures. For example, 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. In other words, quantization could change from one picture to the next. H.263 therefore discloses or suggests the *default level of quantization is updated during the encoding of the digital video sequence* (e.g., using PQUANT with each picture).

300. As I described in §X.B.2.a(iv) and §X.B.2.a(v) (for 1[c] and 1[d], respectively), PQUANT is included in each picture header. And as I described in §X.B.2.a(v) (for 1[d]), PQUANT is provided to a decoding process. H.263 therefore further discloses or suggests *providing an indication of the updated default level of quantization to a decoding process*.

- e. **Claim 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.**

301. H.263 discloses or suggests claim 5's additional limitation.

302. As I described in §X.B.2.a(iv) (for 1[c]), H.263's PQUANT defines a *default level of quantization*. H.263 explains that PQUANT "indicates the quantizer QUANT to be used for the picture until updated by any subsequent GQUANT or

DQUANT.” H.263, 30. In other words, GQUANT (specified at the GOB layer) and/or DQUANT (specified at the macroblock layer) can update (i.e., change) the value of QUANT—used for quantizing the blocks within a picture—from a value previously specified for that picture (e.g., by PQUANT, specified at the picture layer). H.263, 32-33, 37. Therefore, and as I explain further below, both GQUANT and DQUANT disclose or suggest quantizing *based on a level of quantization different from the default level of quantization* (e.g., different from the quantization specified by PQUANT).

303. GQUANT is part of the GOB header that may be transmitted for a GOB.⁶ H.263, 32, Fig. 9. H.263 explains that GQUANT is:

A fixed length codeword of 5 bits which indicates the quantizer QUANT to be used for the remaining part of 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, 33.

⁶ As I explained in §X.B.2.a(iv), in some circumstances the GOB header may not be transmitted (e.g., for first GOB in each picture) or may be empty (depending on encoder strategy). It would have been obvious that in other circumstances a non-empty GOB header would be transmitted, in which case GQUANT is part of the GOB.

304. In other words, GQUANT specifies the value for QUANT in place of any previously set value (e.g., by PQUANT). This is consistent with how H.263 describes the operation of PQUANT. H.263, 30 (explaining that PQUANT indicates the quantizer QUANT “until updated by any subsequent GQUANT or DQUANT”). That is, GQUANT sets QUANT in place of the value previously set by PQUANT (the *default level of quantization*). GQUANT therefore causes quantizing *based on a level of quantization different from the default level of quantization* (e.g., PQUANT).

305. DQUANT additionally discloses or suggests quantizing *based on a level of quantization different from the default level of quantization*. As I explained in §X.B.2.a(iv), DQUANT is only included in certain types of macroblocks (as indicated by MCBPC). For example, and as illustrated in H.263’s Table 9 (reproduced and annotated below), DQUANT is included in the macroblock type named “INTER+Q.” The INTER+Q block is INTER coded (as indicated by the macroblock having a “Picture type” of “INTER”) and motion compensated (as indicated by the presence of “MVD,” or Motion Vector Data, in the macroblock). H.263, 34, 38.

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.

H.263, Table 9 (annotated).

306. When present (i.e., as indicated by MCBPC, for example when the macroblock type is INTER+Q), DQUANT is part of the macroblock header as a “two-bit code to define a change in QUANT.” H.263, 33-34, 37, Fig. 10. That is, DQUANT changes QUANT from a previously-specified value (e.g., as indicated by PQUANT). DQUANT therefore additionally discloses or suggests quantizing *based on a level of quantization different from the default level of quantization* (e.g., PQUANT).

307. H.263 additionally discloses or suggests the additional limitations of claim 5.

308. As I described in §X.B.2.a(iii) (for 1[b]), H.263 discloses producing *sets of transform coefficient values*. As I additionally described in §X.B.2.a(iii) (for 1[b]), the *sets of transform coefficient values* are produced by applying a transform

coding technique to blocks of prediction error values, and are therefore *representative of said blocks of prediction error values*. H.263 further explains that “[t]ransmitted blocks are transformed and resulting coefficients are quantized.” H.263, 11. In other words, H.263 quantizes the transform coefficients, which are representative of prediction error values, to produce quantized coefficients. And as described in §X.B.2.a(iv), H.263’s QUANT indicates the quantizer used for quantization. Further, GQUANT “indicates the quantizer QUANT,” and DQUANT “define[s] a change in QUANT.” H.263, 33, 37. That is, the quantization of the transform coefficients (representative of blocks of prediction error values) would be based on any change to QUANT indicated by GQUANT or DQUANT. H.263 therefore discloses or 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 (QUANT) based on GQUANT or DQUANT.

f. Claim 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.

309. H.263 discloses or suggests claim 6’s additional limitation.

310. As I described in §X.B.2.e (for claim 5), H.263’s DQUANT defines a change to a previously specified (e.g., by PQUANT) QUANT (*default level of quantization*). In other words, as I further described in §X.B.2.e, DQUANT

indicates a level of quantization different from the *default level of quantization*. The QUANT changed (from the *default level of quantization*) by DQUANT is *said level of quantization*. DQUANT is itself a differential value (*indication of a difference*).

311. H.263 explains that “DQUANT is a two-bit code to define a change in QUANT. In Table 12 the differential values for the different codewords are given.”

H.263, 37. H.263’s Table 12, referenced above, is reproduced below:

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, Table 12.

312. H.263 further explains that “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. In other words, DQUANT specifies a differential value (-2, -1, 1, or 2) that is added to QUANT, thereby changing QUANT. As a result, DQUANT specifies the difference between a prior value of QUANT (to which the differential value is added) and the new value of QUANT (the result of adding the differential value). Furthermore, the prior value of QUANT may have been set previously by PQQUANT (as I described in, for example, §X.B.2.e for claim 5).

313. H.263 explains that there are circumstances in which the GOB header may not be transmitted (e.g., for first GOB in each picture) or may be empty (depending on encoder strategy). H.263, 32. Because no GOB header is transmitted for first GOB of a picture, and since the encoder could elect (depending on its strategy) to transmit empty GOB headers for subsequent GOBs, there are situation in which no non-empty GOB headers—and no GQUANT—is coded for a picture. In other words, situations in which no GQUANT is coded for a picture after the transmission of PQUANT (specifying QUANT) for that picture.

314. H.263's Figure 6, part 1 of 7 (illustrating the syntax of the picture layer), part 3 of 7 (illustrating the syntax of the GOB layer), and part 7 of 7 (illustrating the syntax of the macroblock layer), has been reproduced and annotated below to further illustrate the above. In particular, the annotated figure shows how at the GOB layer, no GQUANT (green) may be included in the video bitstream (blue path) (e.g., a GOB header is not transmitted or is empty), such that when DQUANT is coded at the macroblock layer (purple), it changes the value of QUANT previously set by PQUANT for that picture (red) (i.e., it changes *the default level of quantization*). DQUANT therefore discloses or suggests *an indication of a difference between said level of quantization and the default level of quantization*.

317. As I described in §X.B.2.e (for claim 5), H.263's DQUANT, specified at the macroblock layer, defines a change to a previously specified QUANT. As I further described in §X.B.2.a(iv) and §X.B.2.e, the previously specified QUANT may have been provided by H.263's PQUANT, specified at the picture layer. The previously specified QUANT (e.g., by PQUANT) is therefore a *default level of quantization*. And further, the QUANT, as 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*.

318. As I described in §X.B.1 (H.263 overview), H.263's 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 shown in Figure 6 (parts 1 and 3 of 7) from H.263, which has been reproduced and annotated below, the top picture layer of a video bitstream includes PQUANT (in red) for a picture. Under the picture layer is the GOB layer data for all of the GOBs that form the picture (in blue). Each of the GOBs may include a GOB header (as I described in §X.B.2.a(iv) and §X.B.2.f, for 1[c] and claim 6, respectively), and then one layer below include the macroblock layer data for all macroblocks of the GOB (in purple). The syntax repeats (in green) for the next picture in the video.

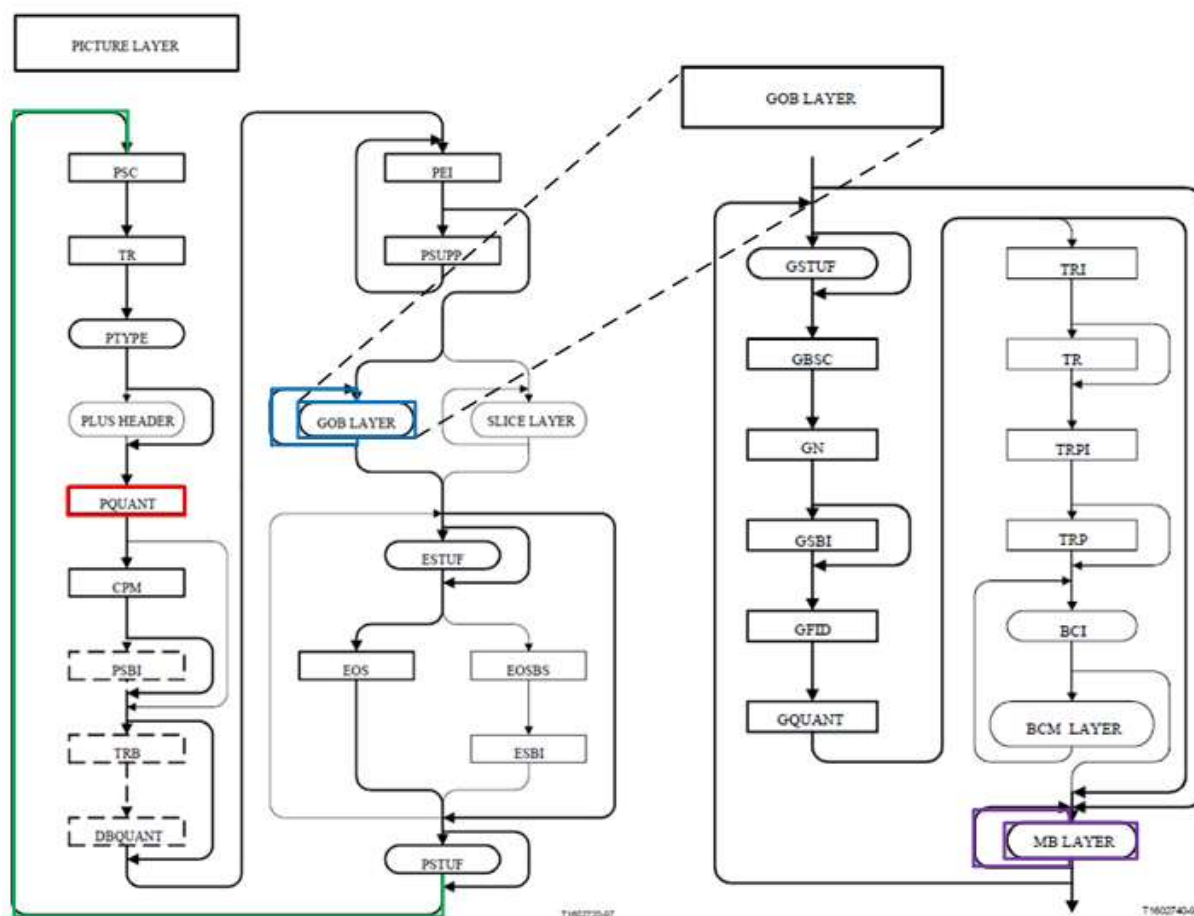


Figure 6/H.263 (part 1 of 7) – Syntax diagram for the video bitstream

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

H.263, Fig. 6 (parts 1 and 3 of 7) (annotated).

319. That is, for every picture in the video there is picture layer data, which includes GOB layer and macroblock layer data for all GOBs and macroblocks within the picture. As a result, PQUANT is transmitted with each picture of the video, which specifies QUANT for that picture (*default level of quantization*). Then, when the first macroblock of the first GOB of the picture includes DQUANT, DQUANT changes the QUANT that had been specified by PQUANT to a new QUANT (*an actual level of quantization that is different from the default level of*

quantization).⁷ Furthermore, the new QUANT (changed by DQUANT) may be applied to the entire picture to which the macroblock belongs (*applied to the sets of transform coefficients for a particular frame*). This will occur when no other changes to QUANT are specified for the picture (for example, when no GOBs within the picture specify GQUANT (e.g., the GOB headers are empty), and no other macroblocks within the picture specify DQUANT).

320. Thus, a first DQUANT of a first picture (e.g., specified by the first macroblock of the first GOB of the first picture) sets *said level of quantization* for the first picture, and a second DQUANT of a second picture (e.g., specified by the first macroblock of the first GOB of the second picture) sets *said level of quantization* for the second picture. In other words, DQUANT of different pictures can specify picture-level quantization changes (*adjusted from one frame of the digital video sequence to another*). Furthermore, PQUANT of the first and second picture may be set to the same value, such that both specify the same “default level of quantization” subsequently adjusted by DQUANT.

321. H.263 therefore discloses or suggests a level of quantization that is

⁷ When the macroblock in question is part of the first GOB, no GOB header is transmitted. H.263, 32. Therefore no GQUANT could have been transmitted after PQUANT for the picture and before DQUANT of the macroblock in question.

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.

- h. Claim 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.**

322. H.263 discloses or suggests claim 8's additional limitation.

323. As I described in §X.B.2.g (for claim 7), H.263's DQUANT updates QUANT to an *actual level of quantization*. And as I described in §X.B.2.f (for claim 6), DQUANT specifies a differential value added to the prior QUANT (e.g., as set by PQANT) (*representable as a difference with respect to the default level of quantization*).

- i. Claim 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.**

324. H.263 discloses or suggests claim 9's additional limitation.

325. As I described in §X.B.2.g and §X.B.2.h (for claims 7 and 8, respectively), H.263's DQUANT is *an indication of a difference between said actual level of quantization and the default level of quantization*. And as I described in §X.B.2.f (for claim 6), DQUANT is *provid[ed] to a decoding process*.

- j. **Claim 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.**

326. H.263 discloses or suggests claim 10's additional limitation. In particular, and as I describe below, both GQUANT and DQUANT disclose or suggest the additional limitation of claim 10.

327. As I described in §X.B.2.e (for claim 5), both GQUANT (specified at the GOB layer) and DQUANT (specified at the macroblock layer) change QUANT from its previous value (e.g., as specified by PQUANT) (*the default level of quantization*). As I additionally described in §X.B.2.e, the updated QUANT, whether changed by GQUANT or DQUANT, is an *actual level of quantization applied to the sets of transform coefficients* that is *different from the default level of quantization*. Furthermore, as I described in §X.B.2.a(i) (for 1[pre]), H.263's luminance blocks and chrominance blocks (*blocks of pixels*) combine to form macroblocks, which further combine to form GOBs, of a picture (*frame*). Thus, the luminance blocks and chrominance blocks are *blocks of pixels, into which a frame of said digital video sequence is divided*.

328. Describing the composition of macroblocks, H.263 explains how:

Unless in RRU mode, 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 . Further, a macroblock consists of four luminance blocks and the two spatially corresponding colour difference blocks as shown in Figure 5.

H.263, 11.

329. H.263's Figure 5 is reproduced below:

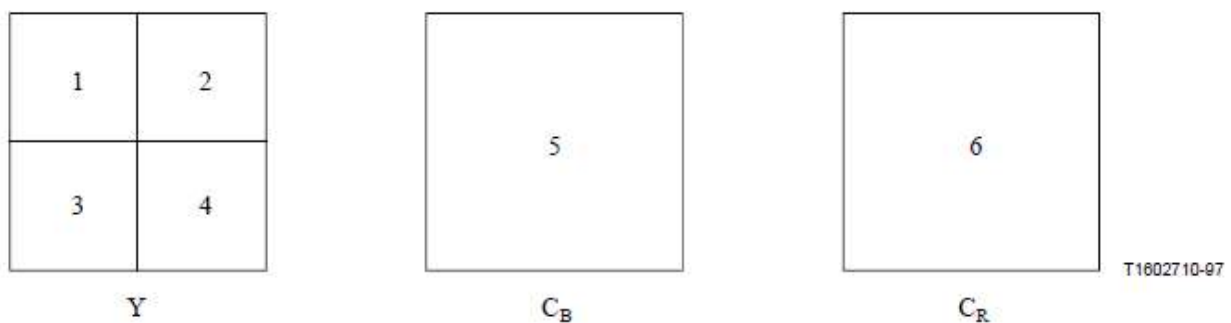


Figure 5/H.263 – Arrangement of blocks in a macroblock

H.263, Fig. 5.

330. Each macroblock is therefore a *segment* into which a *plurality of blocks of pixels* (e.g., four luminance blocks and two chrominance blocks) are *grouped*.

331. As I described in §X.B.2.e and §X.B.2.f (for claims 5 and 6, respectively), DQUANT is part of a macroblock header. Different macroblocks may therefore include DQUANT, each of which updates QUANT. Macroblocks (each a *segment*) that each include DQUANT to update QUANT (*actual level of quantization*) therefore may indicate different values of QUANT that are used for the associated macroblocks (*level of quantization that is adjusted from one segment*

of a frame to another).

332. H.263's GQUANT also discloses or suggests the additional limitation of claim 10. H.263 explains that "[e]ach GOB is divided into macroblocks," and that "a GOB comprises one macroblock row for sub-QCIF, QCIF and CIF, two macroblock rows for 4CIF and four macroblock rows for 16CIF." H.263, 11. Further, sub-QCIF, QCIF, CIF, 4CIF, and 16CIF represent the five standardized picture formats used by H.263. H.263, 7. In other words, a GOB comprises between one and four macroblock rows, depending on the source picture format. Each GOB is therefore also a *segment* into which a *plurality of blocks of pixels* (e.g., a plurality of macroblocks, each of which group four luminance blocks and two chrominance blocks) *are grouped*. And as I described in §X.B.2.e (for claim 5), GQUANT is part of a GOB header. As a result each GOB (*segment*) that includes GQUANT to update QUANT (*level of quantization*) enables different values of QUANT for different GOBs (*adjusted from one segment of a frame to another*).

333. Furthermore, as I described in §X.B.2.a(iv) (for 1[c]) and §X.B.2.e (for claim 5), the updated QUANT (*actual level of quantization that is for a particular segment of a frame*), whether updated by DQUANT or GQUANT for a macroblock or GOB, respectively, is used to quantize transform coefficient values (*applied to the sets of transform coefficients*).

- k. **Claim 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.**

334. H.263 discloses or suggests claim 11's additional limitation.

335. As I described in §X.B.2.j and §X.B.2.f (for claims 10 and 6, respectively), 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

a. Claim 12

- (i) **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:**

336. If the preamble is limiting, H.263 discloses or suggests 12[pre].

337. As I described in §X.B.2.a(i) (for 1[pre]), 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.* 12[pre] recites the same.

338. H.263's Figure 3 (reproduced below) illustrates an H.263 "Source coder." As illustrated in Figure 3, the source coder receives video input data

(“Video in”) and generates video data outputs that are passed to a “video multiplex coder.” Figure 3 illustrates the H.263 source coder “in generalized form,” with main elements that include “prediction, block transformation and quantization.” H.263, 9.

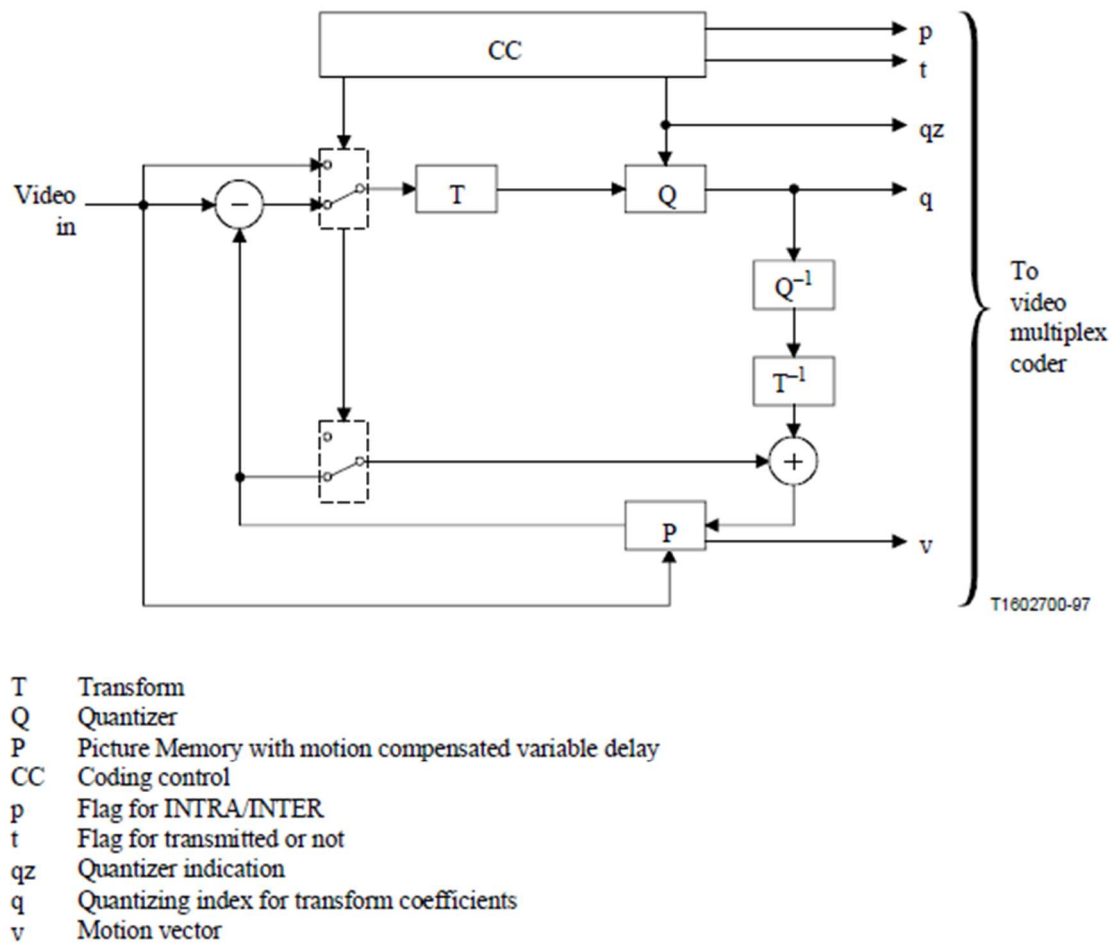


Figure 3/H.263 – Source coder

H.263, Fig. 3.

339. Figure 1 of H.263 (reproduced below) illustrates how the Source coder (e.g. of H.263’s Figure 3) operates as part of a H.263 video codec (i.e., a coder/decoder). As illustrated in Figure 1, the Source coder received a “Video signal,” and generates data that is passed to a “Video multiplex coder” (e.g., as

referenced in Figure 3) and “Transmission buffer,” before producing a “Coded bitstream.” The coded bitstream represents an encoding of the received video signal.

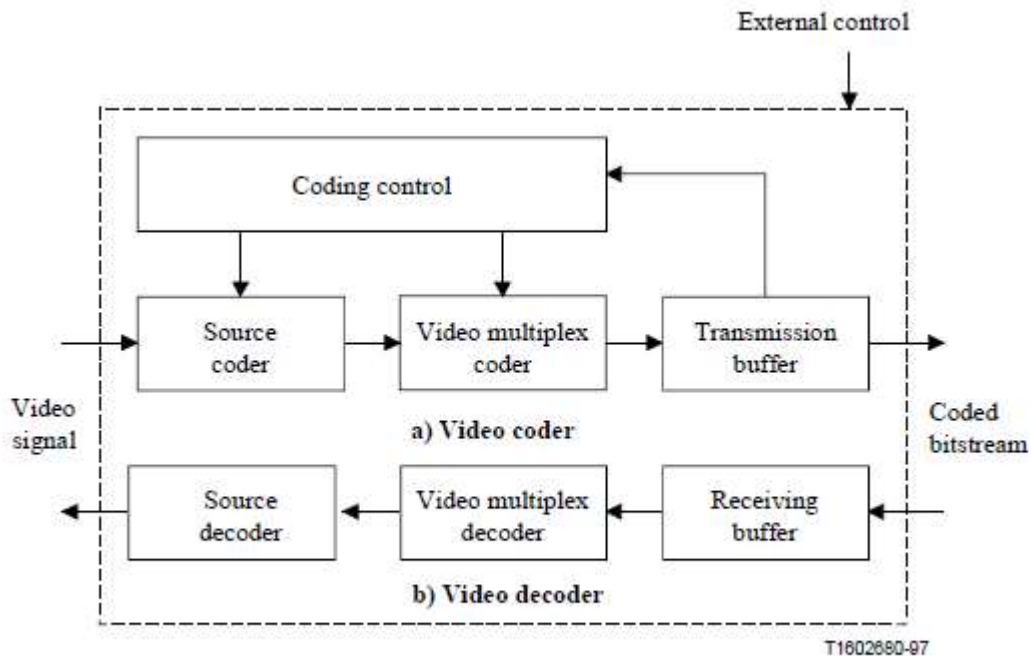


Figure 1/H.263 – Outline block diagram of the video codec

H.263, Fig. 1.

340. The illustrated H.263 source coder is therefore a *video encoder for encoding a digital video sequence*.

- (ii) **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;**

341. H.263 discloses or suggests 12[a].

342. As I described in §IX.A.1 (MPEG-1 overview), 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*. The corresponding structure is a motion compensated predictor and combiner, as explained by the '148 Patent. '148Pat., 7:66-8:4 (motion compensated predictor), 8:5-9 (combiner).

343. As I described in §X.B.2.a(ii) (for 1[a]), H.263 discloses *encoding a frame of the digital video sequence by applying motion compensated prediction to blocks of pixels*. H.263's Figure 3, which illustrates the "main elements" of the H.263 source coder, is reproduced and annotated below. The figure illustrates that the source coder includes a "Picture Memory with motion compensated variable delay" (annotated green), which generates motion vectors (labeled 'v') (annotated light blue) based on video input data (annotated orange). The "Picture Memory with motion compensated variable delay" generates additional data (annotated purple) that is not labeled in the figure. However, a POSITA would have understood that a "Picture Memory with motion compensated variable delay" performs motion compensation (e.g., using the generated motion vectors). For example, H.263's Figure 3 is introduced under the heading "4.2 Video source coding algorithm" (H.263, 9), and under the sub-heading "4.2.2 Prediction," H.263 discusses how "[t]he primary form of prediction is inter-picture and may be augmented by motion compensation." H.263, 11. In other words, the discussion of inter-picture prediction augmented by motion compensation relates to H.263's illustrated source coder.

Therefore, a POSITA would have understood that this additional data (annotated purple) is a prediction formed by performing motion compensation on a reference frame (e.g., as stored in the “Picture Memory”) using the generated motion vectors.

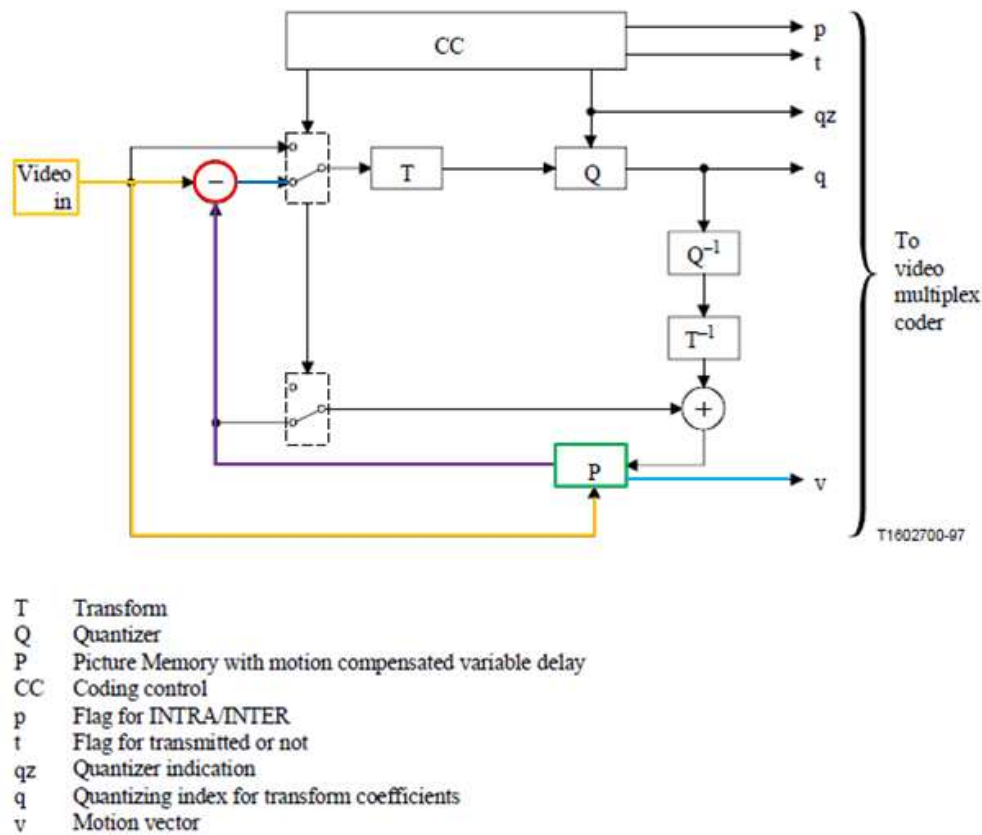


Figure 3/H.263 – Source coder

H.263, Fig. 3 (annotated).

344. Descriptions of H.261, on which H.263 is based (H.263, i), further illustrate that the “Picture Memory with motion compensated variable delay” of H.263’s source coder generates motion compensated predictions. H.261’s Figure 3, reproduced below, illustrates the H.261 source coder, which also includes a “Picture memory with motion compensated variable delay.”

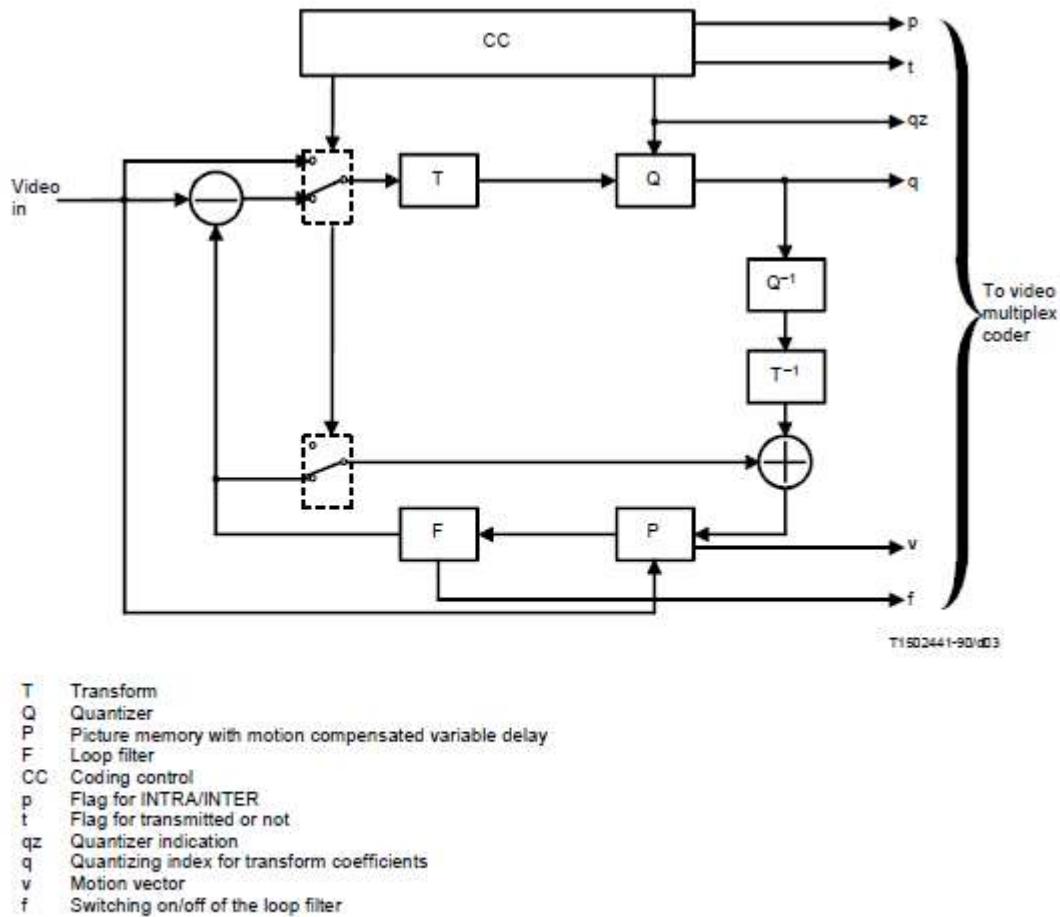


FIGURE 3/H.261
Source coder

H.261, Fig. 3.

345. Other discussions of H.261's encoder illustrate the functionality of the "Picture memory with motion compensated variable delay." For example, Bhaskaran illustrates the "block diagram of an H.261 video encoder," in which the "main elements are frame prediction, DCT transformation, quantization (Q), and variable length coding (VLC)." Bhaskaran, 199. Figure 7.2 of Bhaskaran, which illustrates an H.261 encoder, is reproduced below. The figure includes a "Motion Estimation" block, a "Frame Memory," and a "Motion Compensation" block. A

POSITA would have understood that the “Motion Estimation” block generates motion vectors, used by the “Motion Compensation” block to form motion compensated predictions (the illustrated “Predictive frame”) based on a reference frame (e.g., from the “Frame Memory”).

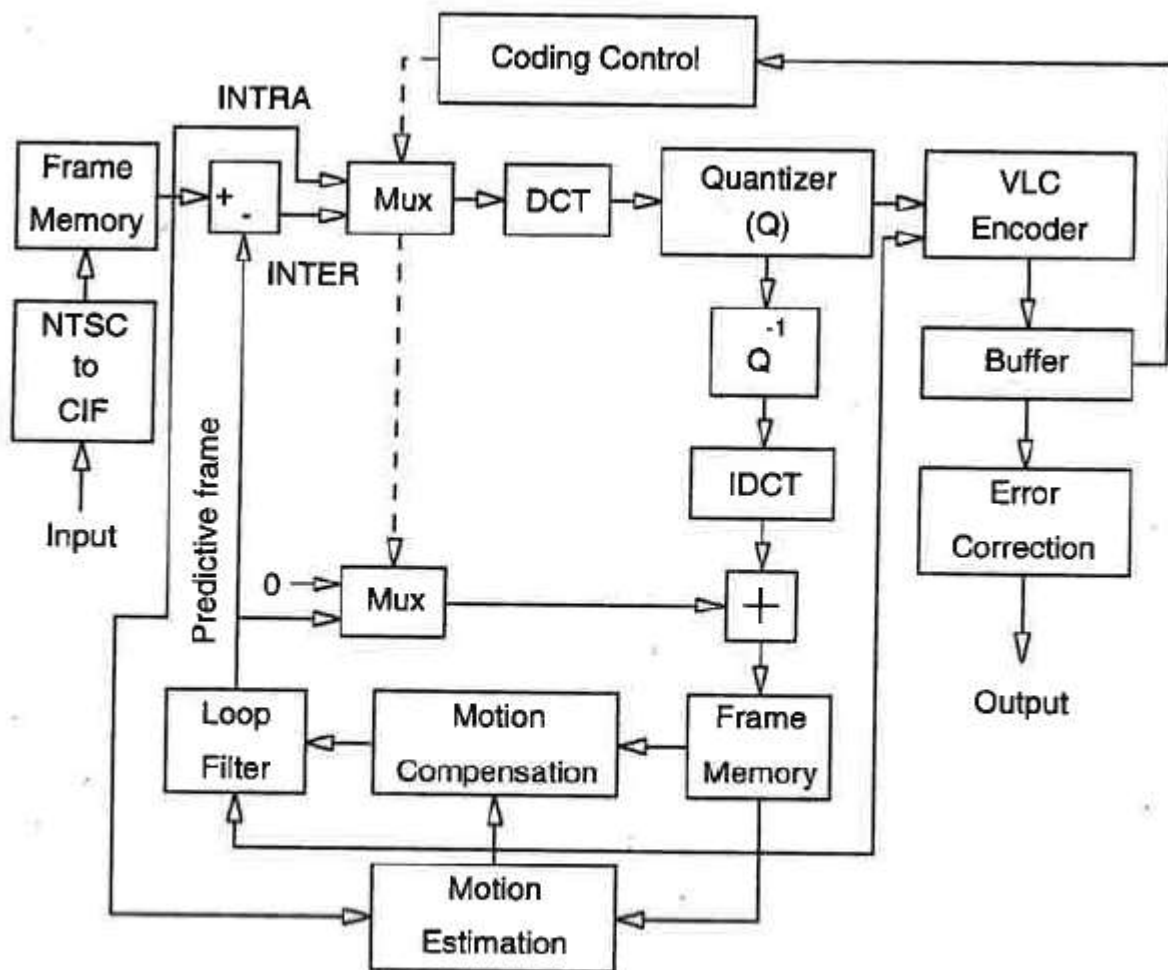


Figure 7.2 Block diagram of an H.261 video encoder.

Bhaskaran, Fig. 7.2.

346. Further, I note that the “Motion Estimation” block, “Motion

Compensation” block, and “Frame Memory” shown in Bhaskaran’s illustration of an H.261 video encoder are used in place of the “Picture memory with motion compensated variable delay” of H.261’s illustrated source coder. In other words, H.261’s source coder illustrates a “Picture memory with motion compensated variable delay,” which (as illustrated by Bhaskaran) encompasses motion estimation, motion compensation, and a frame memory.

347. H.263’s source coder (as illustrated by Figure 3 of H.263) includes the identically-named “Picture Memory with motion compensated variable delay,” and as I described above in addressing this claim limitation, H.263 is based on H.261. A POSITA would therefore have understood the “Picture Memory with motion compensated variable delay” of H.263’s source coder to include the same general functionality of the “Picture memory with motion compensated variable delay” of H.261’s source coder (e.g., motion estimation, motion compensation, and a frame memory). A POSITA therefore would have understood that the “Picture Memory with motion compensated variable delay” of H.263’s source coder is a motion compensated predictor that *app[lies] motion compensated prediction to blocks of pixels* in the same manner as the motion compensated predictor described in the ’148 Patent.

348. H.263 also discloses *provid[ing] corresponding blocks of prediction error values* for the same reasons I described in §X.B.2.a(ii) (for 1[a]). H.263’s Figure 3 (annotated), is reproduced again below. The figure illustrates how H.263’s

source coder takes the difference (annotated red) between video input data (annotated orange) and the result (e.g., a prediction) (annotated purple) of motion-compensated prediction (annotated green). As I described in §X.B.2.a(ii) (for 1[a]), the resulting difference (annotated blue) is *blocks of prediction error values*.

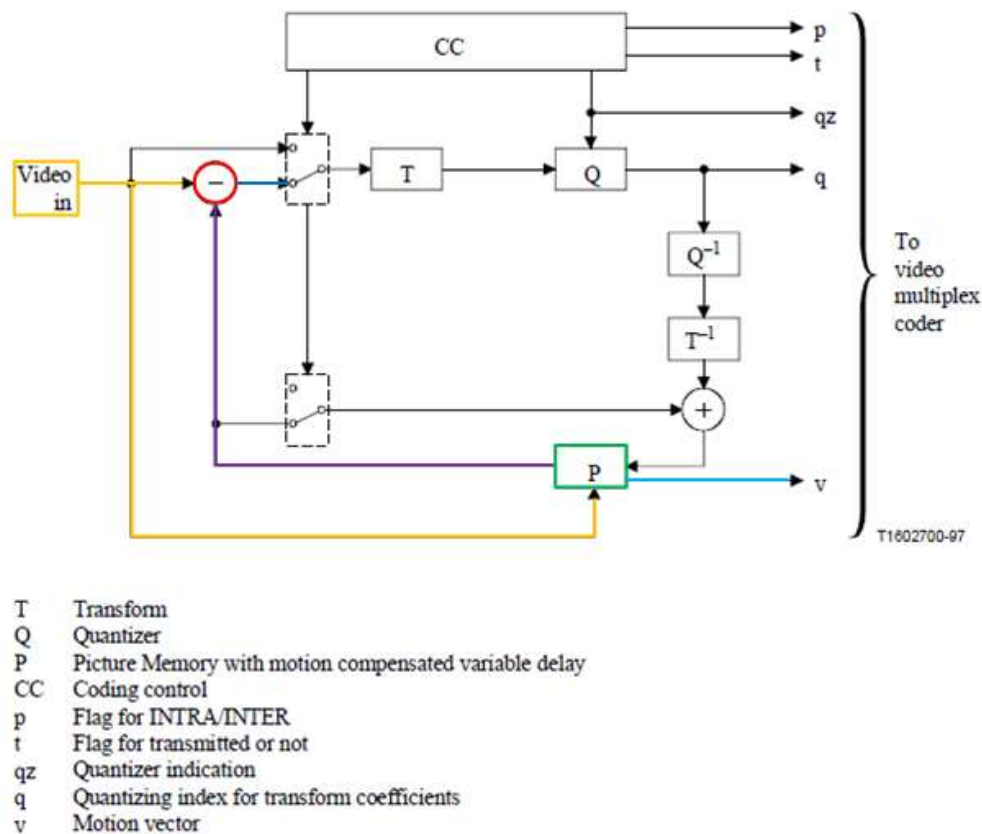


Figure 3/H.263 – Source coder

H.263, Fig. 3 (annotated).

349. In describing the manner in which prediction error values are generated, the '148 Patent explains that:

Each 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.

350. In other words, the '148 Patent describes how a combiner subtracts predicted pixel values from input pixel values to generate prediction error values. In a similar manner, H.263's source coder illustrates how the difference block (red) subtracts predicted pixel values (purple) from input pixel (i.e., video input data) (orange). That is, H.263's difference block operates in the same manner as the combiner of the '148 Patent.

351. A POSITA therefore would have understood that the *blocks of prediction error values* (which H.263 discloses, as I described in §X.B.2.a(ii) for 1[a]), are produced (*provide[d]*) in the H.263 source coder by a combiner (i.e., the difference block, annotated red) in the same manner as the combiner described by the '148 Patent.

(iii) 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;

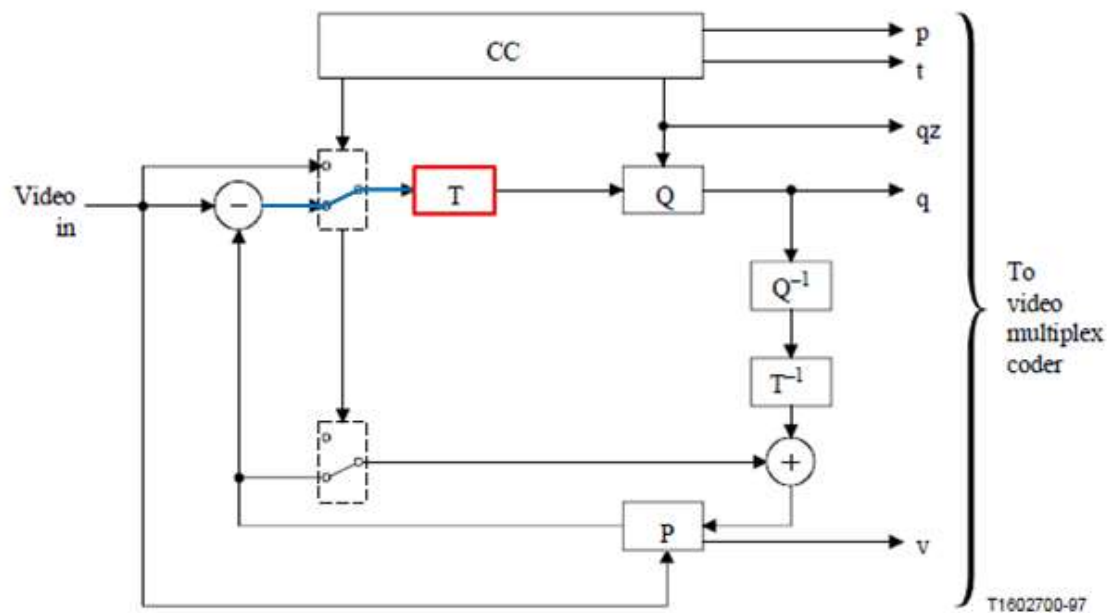
352. H.263 discloses or suggests 12[b].

353. As I described in §IX.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, as explained by the '148 Patent. '148Pat., 18:66-19:3.

354. H.263 discloses or suggests *transforming said blocks of prediction error values for providing sets of transform coefficient values representative of said blocks of prediction error values*, as recited by 12[b], for the reasons I described in §X.B.2.a(iii) for 1[b] and the substantially similar limitation discussed therein. That is, 12[b] recites *transforming* (“applying a transform coding technique to” of 1[b]) *said blocks of prediction error values* (as also recited by 1[b]) *for providing sets of transform coefficient values* (“to produce sets of transform coefficient values” of 1[b]) *representative of said blocks of prediction error values* (as also recited by 1[b]), which H.263 discloses or suggests.

355. H.263’s Figure 3, which illustrates the “main elements” of the H.263 source coder, is reproduced and annotated below. The figure illustrates how the source coder (*video encoder*) includes a “Transform” block (annotated red). The Transform block receives an input (annotated blue), which as I described in §X.B.3.a(ii) (for 12[a]) is *blocks of prediction error values*. A POSITA would have understood that H.263’s Transform block “*provid[es] sets of transform coefficient values*” from the received *blocks of prediction error values* in the same manner as the transform block described in the '148 Patent.



- 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

H.263, Fig. 3 (annotated).

- (iv) **12[c]: means for selecting a default level of quantization for quantizing said sets of transform coefficient values; and**

356. H.263 discloses or suggests 12[c].

357. As I described in §IX.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, as explained by the '148 Patent. '148Pat., 6:16-22; 19:60-65.

358. H.263 discloses or suggests *selecting a default level of quantization for quantizing said sets of transform coefficient values*, as recited by 12[c], for the reasons I described in §X.B.2.a(iv) for 1[c] and the substantially similar limitation discussed therein. That is, 12[c] recites *selecting a default level of quantization* (“defining a default level of quantization” of 1[c]) *for quantizing* (“for use in encoding of the digital video sequence to quantize” of 1[c]) *said sets of transform coefficient values* (“the sets of transform coefficient values” of 1[c]), which H.263 discloses or suggests.

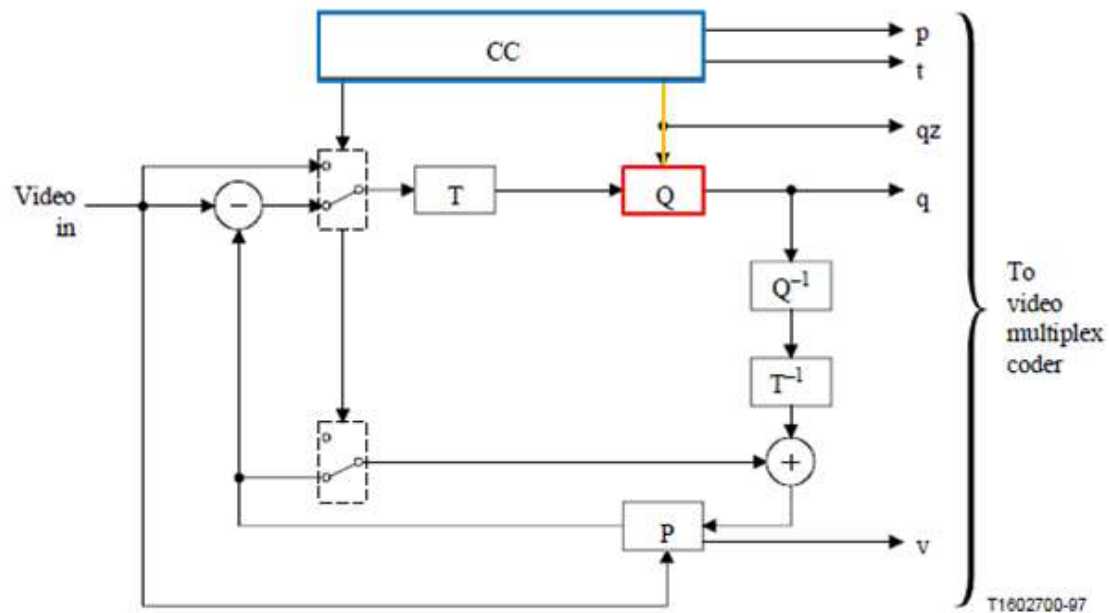
359. H.263 explains, under the heading “Coding control,” how:

Several parameters may be varied to control the rate of generation of coded video data. These include processing prior to the source coder, the quantizer, block significance criterion and temporal subsampling.

H.263, 13.

360. In other words, one of the functions of “Coding control” in H.263 is to vary quantizer parameters to control the coded video rate.

361. Similarly, H.263’s Figure 3 (reproduced and annotated below), which illustrates the “main elements” of the H.263 source coder (*video encoder*), shows that the source coder includes a “Coding control” block (annotated blue), which provides a “Quantizer indication” (also labeled “qz” and annotated orange) to a “Quantizer” block (annotated red).



- 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

H.263, Fig. 3 (annotated).

362. H.263 therefore both explains that one of the functions of “Coding control” is to vary parameters (e.g., quantization parameters) used by a quantizer, and illustrates a “Coding control” block that provides a “Quantizer indication” to a “Quantizer” block. It therefore would have been obvious to a POSITA that the illustrated “Coding control” block would be used to vary (*select*) quantization parameters, including a *default level of quantization*. The illustrated “Coding control” block of H.263 therefore operates in the same manner, and performs the

same functions, of the control manager described in the '148 Patent.

- (v) **12[d]: means for providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence.**

363. H.263 discloses or suggests 12[d].

364. As I described in §IX.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, as explained by the '148 Patent. '148Pat., 6:38-47; 19:60-65.

365. H.263 discloses or suggests *providing an indication of said default level of quantization in the encoded bit-stream representative of the digital video sequence*, as recited by 12[d], for the reasons I described in §X.B.2.a(v) for 1[d] and the substantially similar limitation discussed therein. That is, 12[d] recites *providing an indication of said default level of quantization* (“providing an indication of the default level of quantization” of 1[d]) *in the encoded bit-stream representative of the digital video sequence* (“to a decoding process” of 1[d]), which H.263 discloses or suggests.

366. H.263's Figure 3 (reproduced and annotated below), which illustrates the “main elements” of the H.263 source coder (*video encoder*), shows that the “Quantizer indication” (annotated red) generated by the “Coding control” block (annotated blue), both of which I described in §X.B.3.a(iv) for 12[c], is provided to

a “video multiplex coder” (annotated green).

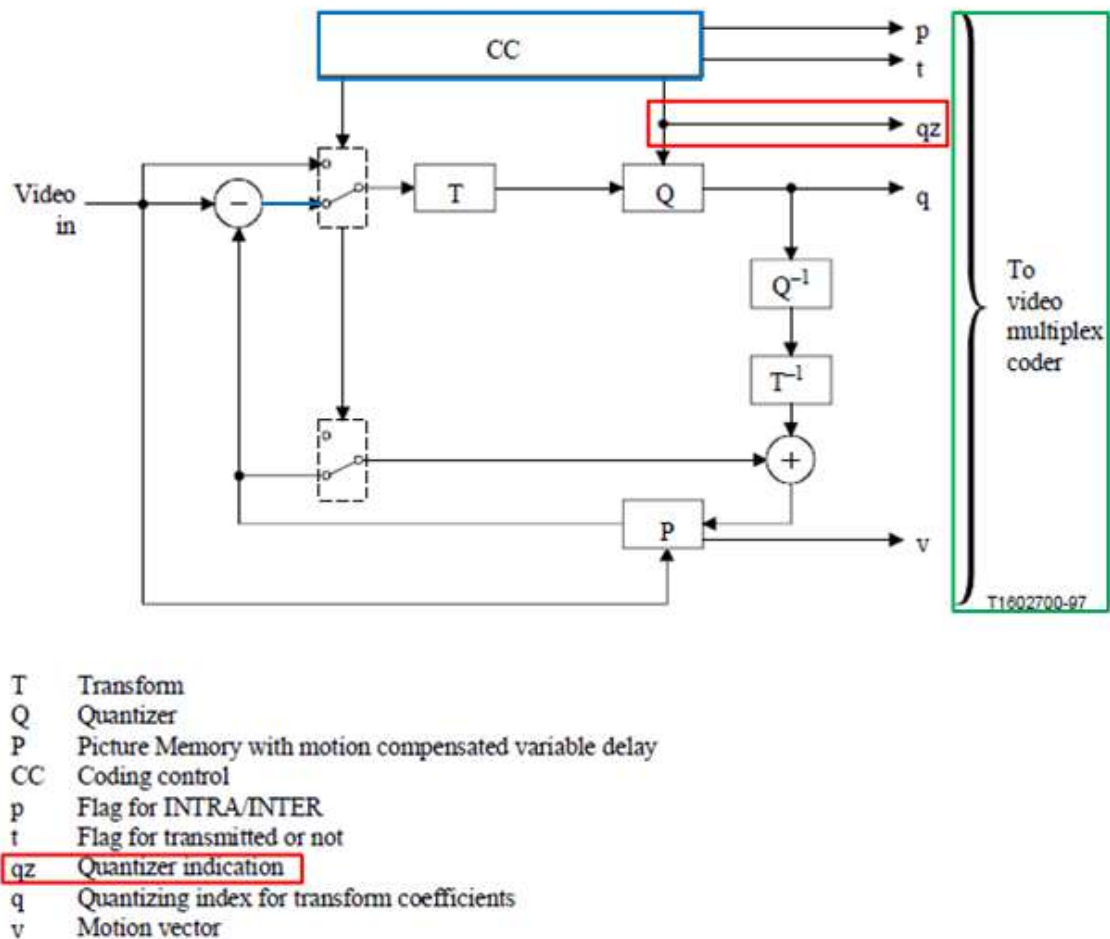


Figure 3/H.263 – Source coder

H.263, Fig. 3 (annotated).

367. Relatedly, Figure 1 of H.263 (reproduced and annotated below) illustrates how a “Video multiplex coder” block (i.e., the video multiplex coder of Figure 3) (annotated red) receives data from a “Source coder” block (i.e., the Source coder of Figure 3), and produces a “Coded bitstream” (annotated purple). That is, the video multiplex coder provides an *encoded bit-stream representative of the digital video sequence* (the “Coded bitstream”).

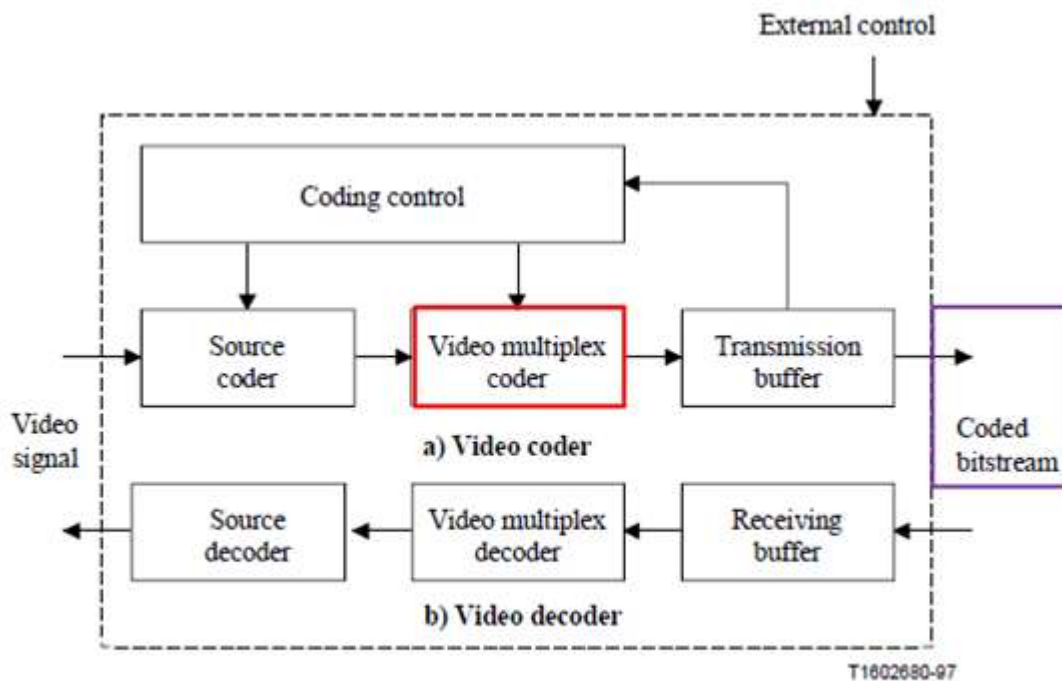


Figure 1/H.263 – Outline block diagram of the video codec

H.263, Fig. 1 (annotated).

368. Therefore, a POSITA would have understood that H.263’s video multiplex coder operates in the same manner as the video multiplex coder described in the ’148 Patent, and provides an *encoded bit-stream representative of the digital video sequence* (the “Coded bitstream”) that includes *an indication of said default level of quantization* (the “Quantizer indication”).

- b. Claim 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.**

369. H.263 discloses or suggests claim 13’s additional limitation.

370. H.263 discloses or 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 same reasons I described in §X.B.2.e for claim 5.

371. Moreover, H.263's Figure 3, which illustrates the source coder (*video encoder*) "in generalized form," is presented as part of the discussion of H.263's "[v]ideo source coding algorithm." H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263's source coder therefore conforms to H.263's encoding method.

c. Claim 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.

372. H.263 discloses or suggests claim 14's additional limitation.

373. H.263 discloses or 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*, as recited by claim 14, for the reasons I described in §X.B.2.f for claim 6 and the substantially similar limitation discussed therein. That is, claim 14 recites *arranged to provide...in the encoded bit-stream* ("providing to a decoding process" of claim 6) *an indication of a difference between said level of quantization and the default level of quantization* (as also recited by claim 6), which

H.263 discloses or suggests.

374. Moreover, H.263's Figure 3, which illustrates the source coder (*video encoder*) "in generalized form," is presented as part of the discussion of H.263's "[v]ideo source coding algorithm." H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263's source coder therefore conforms to H.263's encoding method that provides the indication *in the encoded bit-stream* sent to a decoder.

- d. Claim 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.**

375. H.263 discloses or suggests claim 15's additional limitation.

376. H.263 discloses or 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*, as recited by claim 15, for the reasons I described in §X.B.2.d for claim 4 and the substantially similar limitation discussed therein. That is, claim 15 recites *wherein the default level of quantization can be updated* ("wherein the default level of quantization is updated during the encoding of the digital video sequence" of claim 4) *and the video encoder is arranged to provide an indication of the updated default level of quantization in the encoded bit-stream* ("said method further comprising providing an indication of the updated default level of quantization to a decoding process" of

claim 4), which H.263 discloses or suggests.

377. Moreover, H.263's Figure 3, which illustrates the source coder (*video encoder*) "in generalized form," is presented as part of the discussion of H.263's "[v]ideo source coding algorithm." H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263's source coder therefore conforms to H.263's encoding method that provides the indication *in the encoded bit-stream* sent to a decoder.

- e. **Claim 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.**

378. H.263 discloses or suggests claim 16's additional limitation.

379. As I described in §X.B.3.b for claim 13, 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* (for example, based on GQUANT or DQUANT) *different from* a QUANT that had been set previously. As I further described in §X.B.3.d for claim 15, H.263 additionally discloses an *updated default level of quantization* (for example, based on PQUANT for different pictures specifying different values for QUANT).

380. A POSITA would have understood, based on what I described above,

that PQUANT for different pictures may specify different values for QUANT to be used for that picture, where each picture's QUANT may be changed by GQUANT or DQUANT. Therefore, either GQUANT or DQUANT can change QUANT of a picture to *a level of quantization different from* what had been specified by PQUANT, where the PQUANT of the corresponding picture may have specified a QUANT value different from the PQUANT of another picture (*the updated default level of quantization*).

f. Claim 17: A video encoder according to claim 12, provided in a multimedia terminal.

381. H.263 discloses or suggests claim 17's additional limitation.

382. H.263 "specifies a coded representation that can be used for compressing the moving picture component of audio-visual services at low bit rates." H.263, 1. H.263 further explains that "the standardized source formats on which the source coder operates are based on a Common Intermediate Format (CIF)" in order to "cover use in and between regions using 625- and 525-line television standards." H.263, 2. In other words, H.263 can be used to encode video for different television standards.

383. A POSITA would have understood that when H.263 describes that it "can be used for compressing the moving picture component of audio-visual services" (H.263, 1), the "audio-visual services" (i.e., services that include audio data and visual data) with which H.263 is used includes multimedia data. Relatedly,

the ITU publishes different “Recommendation Series,” and H.263 belongs to “Series H,” titled “Audiovisual and multimedia systems.” H.263, 155. In other words. H.263 relates to audiovisual and multimedia systems.

384. Based on the above, it would have been obvious to a POSITA to provide the H.263 source coder (*video encoder*) in a *multimedia terminal* to, for example, encode the video component of audiovisual and multimedia data. For example, it would have been obvious to use the source coder in a *multimedia terminal* used to encode video for television data.

g. Claim 18: A video encoder according to claim 12, provided in a radio telecommunications device.

385. H.263 discloses or suggests claim 18’s additional limitation.

386. As I described above in §X.B.3.f (claim 17), H.263 describes how it can be used to encode the video component of television data. Furthermore, a POSITA would have known that certain television broadcasts, such as terrestrial television or over-the-air television, transmit television content (e.g., to a receiving television) over radio waves. It therefore would have been obvious to a POSITA to provide an H.263 source coder (*video encoder*), used to encode television data, in a device capable of transmitting the H.263 data stream (e.g., the generated encoded video data) wirelessly over a radio channel (*a radio telecommunications device*).

387. Additionally, H.263 references ITU-T Recommendation H.324 (entitled “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, 1, 8-9. H.324 explains that its scope “covers the technical requirements for very low bit-rate multimedia telephone terminals operating over the General Switched Telephone Network (GSTN),” and that these “H.324 terminals provide real-time video, audio, or data, or any combination, between two multimedia telephone terminals over a GSTN voice band network connection.” H.324, 1. H.324 also explains that these terminals, or “visual telephone system,” may interworks with other terminals “on mobile radio networks.” H.324, 1. A POSITA would have understood that an H.234 terminal, or visual telephone system, interworking with other such systems over a mobile radio network, is *a radio telecommunications device*. Moreover, Figure 1 of H.324 (reproduced below) illustrates that the H.263 “Video codec” (i.e., coder/decoder) may be *provided in* the H.324 terminal.

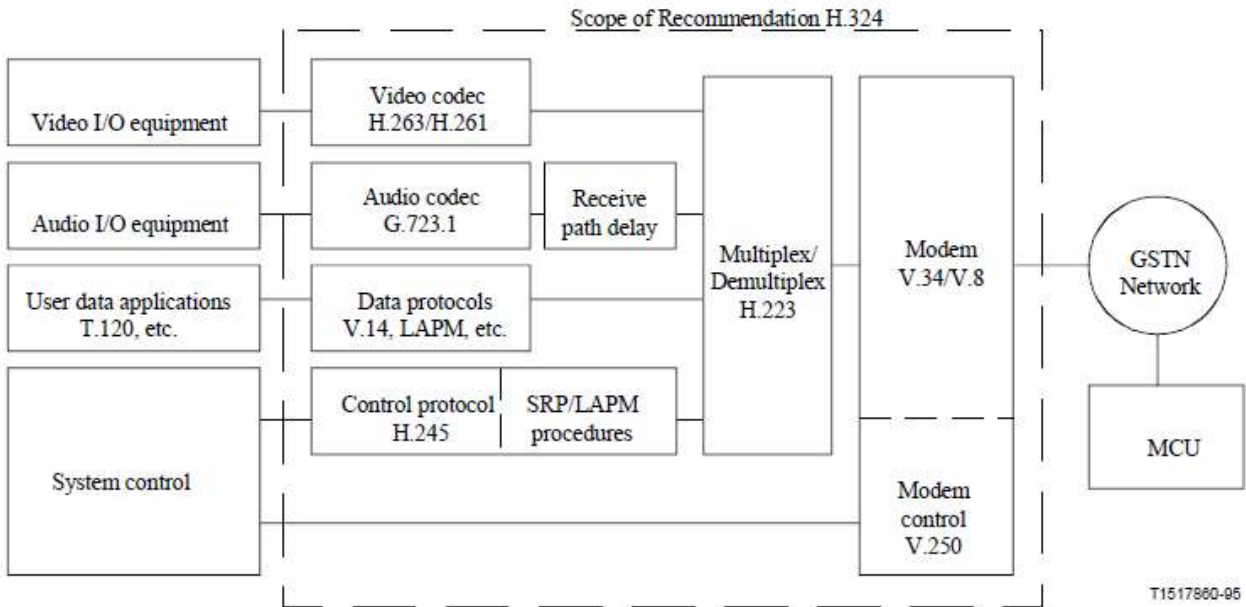


Figure 1/H.324 – Block diagram for H.324 multimedia system

H.324, Fig. 1.

- h. **Claim 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.**

388. H.263 discloses or suggests claim 19's additional limitation.

389. H.263 discloses or 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, as recited by claim 19, for the reasons I described in §X.B.2.g for claim 7 and the substantially similar limitation discussed therein. That is, claim 19 recites *wherein said level of quantization applied to said sets of transform coefficient values* (“wherein said level of quantization” of claim 7) *can be adjusted from one frame of the digital video sequence to another* (“is adjusted from one frame of the digital video sequence to another” of claim 7) *so as to apply an actual level of quantization to the sets of transform coefficients for a particular frame of the digital video sequence* (“such that an actual level of quantization applied to the sets of transform coefficients for a particular frame of the digital video sequence” of claim 7) *and wherein said actual level of quantization is different from the default level of quantization* (“is different from the default level of quantization” of claim 7), which H.263 discloses or suggests.

390. Moreover, H.263’s Figure 3, which illustrates the source coder (*video encoder*) “in generalized form,” is presented as part of the discussion of H.263’s “[v]ideo source coding algorithm.” H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263’s source coder therefore conforms to H.263’s encoding method.

- i. **Claim 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.**

391. H.263 discloses or suggests claim 20’s additional limitation.

392. H.263 discloses or suggests *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*, as recited by claim 20, for the reasons I described in §X.B.2.i for claim 9 and the substantially similar limitations discussed therein. That is, claim 20 recites *arranged to provide an indication...in the encoded bit-stream representative of the digital video sequence* (“further comprising providing to a decoding process an indication” of claim 9) *of said difference with respect to the default level of quantization* (“of a difference between said actual level of quantization and the default level of quantization” of claim 9), which H.263 discloses or suggests.

393. Moreover, H.263’s Figure 3, which illustrates the source coder (*video encoder*) “in generalized form,” is presented as part of the discussion of H.263’s “[v]ideo source coding algorithm.” H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263’s source coder therefore conforms to H.263’s encoding method that provides the indication *in the encoded bit-stream* sent to a decoder.

- j. **Claim 21: 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 to adjust the level of quantization applied to said sets of transform coefficient values from one segment of a frame to another 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, 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.**

394. H.263 discloses or suggests claim 21's additional limitation.

395. 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 both of which I discussed in §X.B.2.j and §X.B.2.k, respectively). 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*”.

Claim 21		Corresponding Elements from Claims 10 and 11
Reference	Claim Limitation	
21[a]	A video encoder according to claim 13, arranged to	wherein the plurality of blocks of pixels, into which a frame of said digital video

	group the plurality of blocks into which a frame of said digital video sequence is divided into one or more segments and	sequence is divided, are grouped into one or more segments; (Claim 10, §X.B.2.j)
21[b]	to adjust the level of quantization applied to said sets of transform coefficient values from one segment of a frame to another	wherein the level of quantization applied to said sets of transform coefficient values is adjusted from one segment of a frame to another; (Claim 10, §X.B.2.j)
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,	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 10, §X.B.2.j)
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.	providing to a decoding process an indication of a difference between said level of quantization and the default level of quantization. (Claim 11, §X.B.2.k)

396. H.263 therefore discloses or suggests claim 21's additional limitation

for the reasons I discussed in in §X.B.2.j and §X.B.2.k.

397. Moreover, H.263's Figure 3, which illustrates the source coder (*video encoder*) "in generalized form," is presented as part of the discussion of H.263's "[v]ideo source coding algorithm." H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263's source coder therefore conforms to H.263's encoding method.

- k. Claim 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.**

398. H.263 discloses or suggests claim 22's additional limitation.

399. H.263 discloses or suggests *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*, as recited by claim 22, for the reasons I described in §X.B.2.k for claim 11, and the substantially similar limitation discussed therein. That is, claim 22 recites *further arranged to provide an indication...in the encoded bit-stream representative of the digital video sequence* ("further comprising providing to a decoding process an indication" of claim 11) *of said difference with respect to the default level of quantization* ("of a difference between said level of quantization and the default level of quantization" of claim 11), which H.263 discloses or suggests.

400. Moreover, H.263's Figure 3, which illustrates the source coder (*video*

encoder) “in generalized form,” is presented as part of the discussion of H.263’s “[v]ideo source coding algorithm.” H.263, 9. In other words, the illustrated source coder performs the described video source coding algorithm. H.263’s source coder therefore conforms to H.263’s encoding method that provides the indication *in the encoded bit-stream* sent to a decoder.

XI. CONCLUSION

401. I currently hold the opinions expressed in this declaration. But my analysis may continue, and I may acquire additional information and/or attain supplemental insights that may result in added observations.

402. I hereby declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct to the best of my knowledge. I understand that willful false statements are punishable by fine or imprisonment or both. *See* 18 U.S.C. § 1001.

Date: June 11, 2025

By: 

Dr. Eli Saber, Ph.D.