UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ELEMENT TV COMPANY, LP AND
ELEMENT TELEVISION COMPANY, LLC,

Petitioners

v.

NOKIA CORPORATION and NOKIA TECHNOLOGIES OY,

Patent Owner.

IPR2025-01108

U.S. Patent No. 7,532,808

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,532,808

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
U.S. Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I. INTRODUCTION 1

II. GROUNDS FOR STANDING 1

III. IDENTIFICATION OF CHALLENGE 1

A. Prior Art 1

B. Grounds for Challenge 2

IV. '808 PATENT OVERVIEW 2

V. LEVEL OF ORDINARY SKILL 10

VI. CLAIM CONSTRUCTION 10

VII. DETAILED EXPLANATION OF GROUNDS 11

A. Ground 1: Obviousness over Karczewicz 11

1. Karczewicz 11

2. Claim 1 and Dependents 13

a. Claim 1 13

(i) 1[pre] 13

(ii) 1[a] 14

(iii) 1[b] 24

(iv) 1[c] 25

(v) 1[d] 25

b. Claim 2 27

c. Claim 6 28

d. Claim 23 29

(i) 23[pre] 29

(ii) 23[a] 29

(iii) 23[b] 30

TABLE OF CONTENTS
(continued)

Page

(iv) 23[c] 31
a. Claim 24 31
b. Claim 25 32
c. Claim 28 32
d. Claim 29 32
e. Claim 30 33
3. Claim 7 and Dependents 35
a. Claim 7 35
(i) 7[pre] 35
(ii) 7[a] 35
(iii) 7[b] 35
(iv) 7[c] 37
b. Claim 9 37
c. Claim 32 38
d. Claim 34 38
e. Claim 36 38
f. Claim 39 38
g. Claim 40 39
h. Claim 41 39
i. Claim 43 39
j. Claim 65 39
B. Ground 2: Obviousness over MPEG-1 and H.263 40

TABLE OF CONTENTS
(continued)

Page

1. MPEG-1 40
2. H.263 42
3. Motivation to combine MPEG-1 and H.263 46
4. Claim 1 and Dependents 49

a. Claim 1 49

(i) 1[pre] 49

(ii) 1[a] 51

(iii) 1[b] 54

(iv) 1[c] 56

(v) 1[d] 57

b. Claim 2 60

c. Claim 6 61

d. Claim 23 64

(i) 23[pre] 64

(ii) 23[a] 64

(iii) 23[b] 65

(iv) 23[c] 65

e. Claim 24 66

f. Claim 25 67

g. Claim 28 67

h. Claim 29 68

i. Claim 30 68

5. Claim 7 and Dependents 69

TABLE OF CONTENTS
(continued)

Page

a. Claim 7 69

(i) 7[pre] 69

(ii) 7[a] 71

(iii) 7[b] 71

(iv) 7[c] 71

b. Claim 9 72

c. Claim 32 73

d. Claim 34 73

e. Claim 36 74

f. Claim 39 74

g. Claim 40 74

h. Claim 41 74

i. Claim 43 75

j. Claim 65 75

VIII. CONCLUSION 75

IX. MANDATORY NOTICES 76

A. Real Party in Interest 76

B. Related Matters 76

C. Notice of Counsel and Service Information 77

EXHIBIT LIST

Exhibit	Reference
1001	U.S. Patent No. 7,532,808 ("'808 Patent" or "'808Pat.")
1002	File History of the '808 Patent ("'808FH")
1003	Declaration of Dr. Eli Saber ("Saber")
1004	Dr. Eli Saber CV
1005	RESERVED
1006	International Patent Publication No. WO 01/11891 A1 ("Karczewicz")
1007	ISO/IEC 11172-2:1993, Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s—Part 2: Video ("MPEG-1")
1008	RESERVED
1009	ITU-T Recommendation H.263 (February 1998) ("H.263")
1010	Martyn J. Riley and Iain E.G. Richardson, *Digital Video Communications*, Artech House, Inc. (1997) ("Riley")
1011	Gary J. Sullivan and Thomas Wiegand, *Rate-Distortion Optimization for Video Compression*, IEEE Signal Processing Magazine, pp. 74-90 (November 1998) ("Sullivan")
1012	JVT of ISO/IEC and ITU-T VCEG – JVT-A003r1, Joint Model Number 1, Revision 1 (JM-1r1), Pattaya, Thailand, 3-7 December 2001
1013	Iain E. G. Richardson, *H.264 and MPEG-4 Video Compression*
1014	Wang et al., *Video Processing and Communications* (Prentice Hall, 2002) ("Wang")
1015	Bhaskaran, Vasudev, Konstantinides, Konstantinos, *Image and Video Compression Standards, Algorithms and Architectures*, (Kluwer Academic Publishers, 1995) ("Bhaskaran")
1016	Declaration of June Ann Munford
1017	Complaint, *Element Television Co. v. Nokia Corp.*, No. 24-cv-04269 (D. Mn.)

Exhibit	Reference
1018	Ghanbari, Mohammed, *Video coding, an introduction to standard codecs*, IEEE 1999 (Ghanbari)

LISTING OF CHALLENGED CLAIMS

Reference	Claim Limitation
Claim 1	
1[pre]	A method of encoding a video sequence, the method comprising:
1[a]	assigning a skip coding mode to a first segment of a first frame of the sequence;
1[b]	assigning either a zero motion vector or a predicted non-zero motion vector for the skip coding mode for the first segment based at least in part on the motion information of a second segment neighboring the first segment; and
1[c]	forming a prediction for the first segment with respect to a reference frame based at least in part on the assigned motion vector for the skip coding mode, wherein the assigned motion vector is one of the zero motion vector and the predicted non-zero motion vector; and
1[d]	providing in an encoded bitstream an indication of the skip coding mode, wherein no further motion vector information for the first segment is coded in the encoded bitstream.
Claim 2	
2	A method according to claim 1, wherein the second segment is a previously encoded segment neighboring the first segment.
Claim 6	
6	A method according to claim 1, wherein if the second segment has zero motion vector, the zero motion vector is assigned for skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.

Reference	Claim Limitation
Claim 7	
7[pre]	A method of decoding an encoded video sequence, the method comprising:
7[a]	receiving an indication of a skip coding mode for a first segment;
7[b]	assigning either a zero motion vector or a predicted non-zero motion vector for the skip coding mode for the first segment based at least in part on the motion information of a second segment neighboring the first segment; and
7[c]	forming a prediction for the first segment with respect to a reference frame based at least in part on the assigned motion vector for the skip coding mode, wherein the assigned motion vector is one of the zero motion vector and the predicted non-zero motion vector.
Claim 9	
9	A method according to claim 7, wherein if a segment in a previously decoded region surrounding the first segment has zero motion vector, the zero-motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.
Claim 23	
23[pre]	A method according to claim 1, wherein if the second segment has a non-zero motion vector,
23[a]	the method further comprising deriving the predicted non-zero motion vector based at least in part on the motion vector of the second segment; and
23[b]	the predicted non-zero motion vector is assigned for the skip coding mode of the first segment; and

Reference	Claim Limitation
23[c]	the prediction for the first segment is formed by a motion compensated prediction with respect to the reference frame based at least in part on the predicted non-zero motion vector.
Claim 24	
24	A method according to claim 1, wherein if the second segment has a zero motion vector and the second segment is predicted using motion-compensated prediction from the reference picture, the zero motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.
Claim 25	
25	A method according to claim 1, wherein if the second segment has a zero motion vector and the second segment is predicted using motion-compensated prediction from a second reference picture immediately preceding the picture second segment belongs to, the zero motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.
Claim 28	
28	A method according to claim 1, further comprising: deriving the predicted non-zero motion vector based at least in part on the motion vector of the second segment.
Claim 29	
29	A method according to claim 1, further comprising: deriving the predicted non-zero motion vector based at least in part on the motion

Reference	Claim Limitation
	vector of the second segment and motion vector of a third segment neighboring the first segment.
Claim 30	
30	A method according to claim 1, wherein no residual information is provided for the first segment in the encoded bitstream.
Claim 32	
32	A method according to claim 7, wherein if the second segment has a zero motion vector, the zero motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.
Claim 34	
34	A method according to claim 7, wherein if the second segment has a non-zero motion vector, the method further comprising deriving the predicted non-zero motion vector based at least in part on the motion vector of the second segment; and the predicted non-zero motion vector is assigned for the skip coding mode of the first segment; and the prediction for the first segment is formed by a motion compensated prediction with respect to the reference frame based at least in part on the predicted non-zero motion vector.
Claim 36	
36	A method according to claim 7, wherein if the second segment has a zero motion vector and the second segment is predicted using motion-compensated prediction from the reference picture, the zero motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.

Reference	Claim Limitation
Claim 39	
39	A method according to claim 7, further comprising: deriving the predicted non-zero motion vector based at least in part on the motion vector of the second segment.
Claim 40	
40	A method according to claim 7, further comprising: deriving the predicted non-zero motion vector based at least in part on the motion vector of the second segment and motion vector of a third segment neighboring the first segment.
Claim 41	
41	A method according to claim 7, wherein no residual information is provided for the first segment in the encoded bitstream.
Claim 43	
43[1]	A method according to claim 7, wherein if the second segment has a zero motion vector and the second segment is predicted using motion-compensated prediction from a second reference picture immediately preceding the picture second segment belongs to, the zero motion vector is assigned to the skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the zero motion vector.
Claim 65	

[1] A Certificate of Correction issued changing the dependency of claim 43 from claim 1 to claim 7. EX-1002, 277.

Reference	Claim Limitation
65	A method according to claim 7, wherein no further motion vector information for the first segment is retrieved from the encoded bitstream.

I. INTRODUCTION

Element TV Company, LP and Element Television Company, LLC ("Element" or "Petitioners") request *inter partes* review ("IPR") of claims 1-2, 6-7, 9, 23-25, 28-30, 32, 34, 36, 39-41, 43, 65 ("Challenged Claims") of U.S. Patent No. 7,532,808 ("'808 Patent"; EX-1001).

II. GROUNDS FOR STANDING

Petitioners certify the '808 Patent is available for IPR and Petitioners are not barred or estopped from requesting IPR on the grounds herein. Petitioners file this Petition within one year of filing a district court complaint against Nokia Corporation and Nokia Technologies Oy (collectively, "Patent Owner" or "PO") seeking declaratory judgment of non-infringement of the '808 Patent. *See Element Television Co. v. Nokia Corp.*, No. 24-cv-04269 (D. Mn.) (EX-1017).

III. IDENTIFICATION OF CHALLENGE

A. Prior Art

Petitioners respectfully request cancellation of the Challenged Claims based on:

Prior Art
Karczewicz (WO 01/11891 A1), published February 15, 2001; prior art under at least 35 U.S.C. §102(b)[2] (EX-1006).
MPEG-1 (ISO/IEC 11172-2:1993, Information technology — Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s—Part 2: Video); published August 1, 1993 and publicly available no later than October 21, 1993; prior art under 35 U.S.C. §102(b) (EX-1007)[3].
H.263; ratified February 6, 1998 and publicly available no later than May 11, 2000[4]; prior art under at least 35 U.S.C. §102(b) (EX-1009).

B. Grounds for Challenge

Ground		Claims	Prior Art
1	§103	1-2, 6-7, 9, 23-25, 28-30, 32, 34, 36, 39-41, 43, 65	Karczewicz
2		1-2, 6-7, 9, 23-25, 28-30, 32, 34, 36, 39-41, 43, 65	MPEG-1, H.263

IV. '808 PATENT OVERVIEW

The '808 Patent is titled "Method for Coding Motion in a Video Sequence" and issued on May 12, 2009. The '808 Patent issued from U.S. Patent Application

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

[3] MPEG-1 was publicly accessible in 1993, more than eight years before the earliest claimed priority date of the '808 Patent (March 15, 2002). EX-1016, ¶24.

[4] H.263 was publicly accessible in May 11, 2000, nearly two years before the earliest claimed priority date of the '808 Patent. EX-1016, ¶19.

10/390,549 (the "'549 Application"), filed March 14, 2003. The '808 Patent claims priority to U.S. Provisional Application 60/365,072 ("'808 Provisional"), filed March 15, 2002. Because all prior art in this petition predates the '808 Provisional's filing date, the Board need not determine whether the '808 Patent's claims are entitled to the priority date of the '808 Provisional.

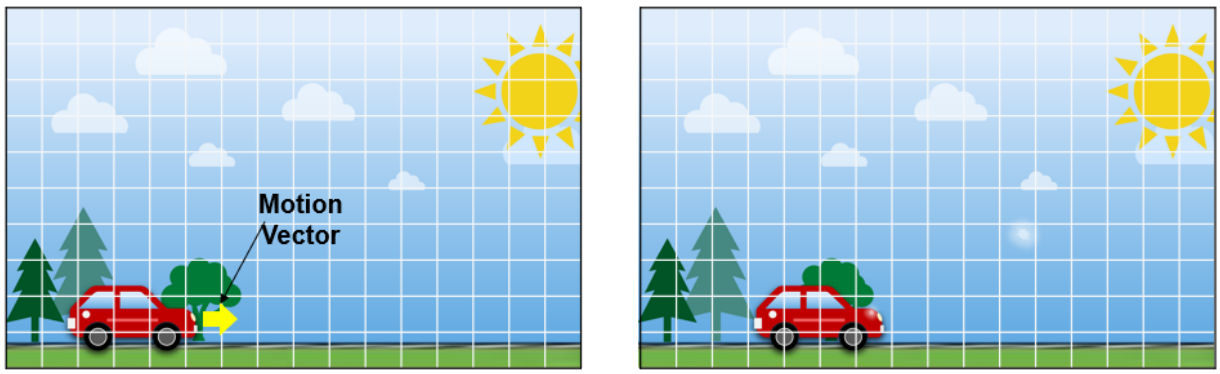
The '808 Patent is directed to video encoding. '808Pat., Abstract. The Background section[5] explains a video sequence is formed of consecutive images, or frames, displayed at a relatively fast rate, creating the illusion of motion. '808Pat., 1:15-19. Because of the large amount of data required to transmit video data in uncompressed form (e.g., as uncompressed frames), various video compression techniques are used to reduce redundancies in a video sequence. '808Pat., 2:7-20, 2:35-37.

One such 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. '808Pat., 3:11-21. Motion-compensated prediction may be performed on individual portions of a frame, known as "macroblocks" (each

[5] Over half of the '808 Patent specification is dedicated to the "Background of the Invention." '808Pat., 3:44-10:22.

macroblock may be predicted based on a motion vector for that macroblock). '808Pat., 1:58-62, 3:23-25. "Frames of a video sequence which are compressed using motion-compensated prediction are generally referred to as INTER-coded." '808Pat., 3:26-28. Further, "[a]ny frame encoded in the INTER-coded format requires a reference frame for motion-compensated prediction." '808Pat., 7:5-6.

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").[6] Saber, ¶48.



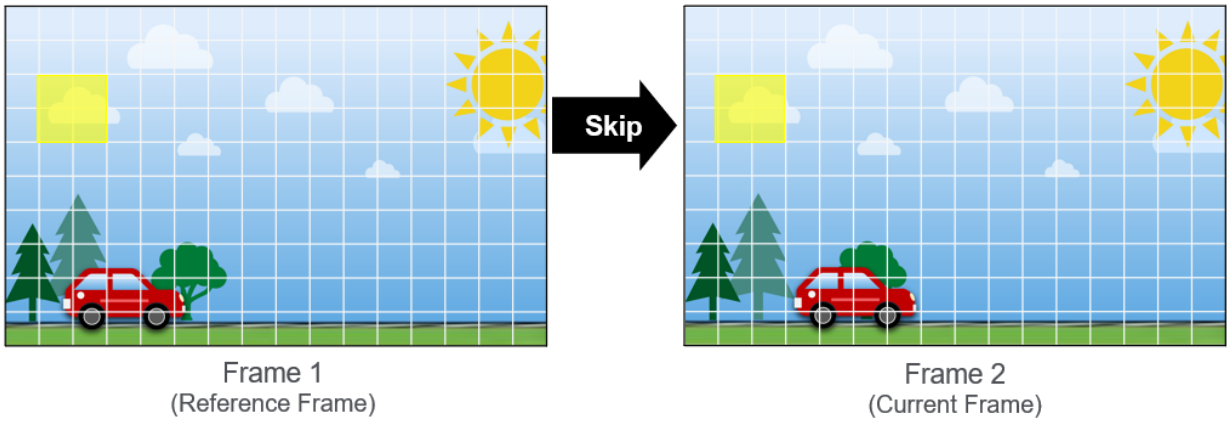
The Background further explains generating a single bitstream (e.g., for transmission to a decoder) that encodes information characterizing the prediction of

[6] All annotations added unless otherwise noted.

frames and macroblocks therein (e.g., motion vector information, prediction error), and related control information. '808Pat., 4:6-12, 6:37-44.

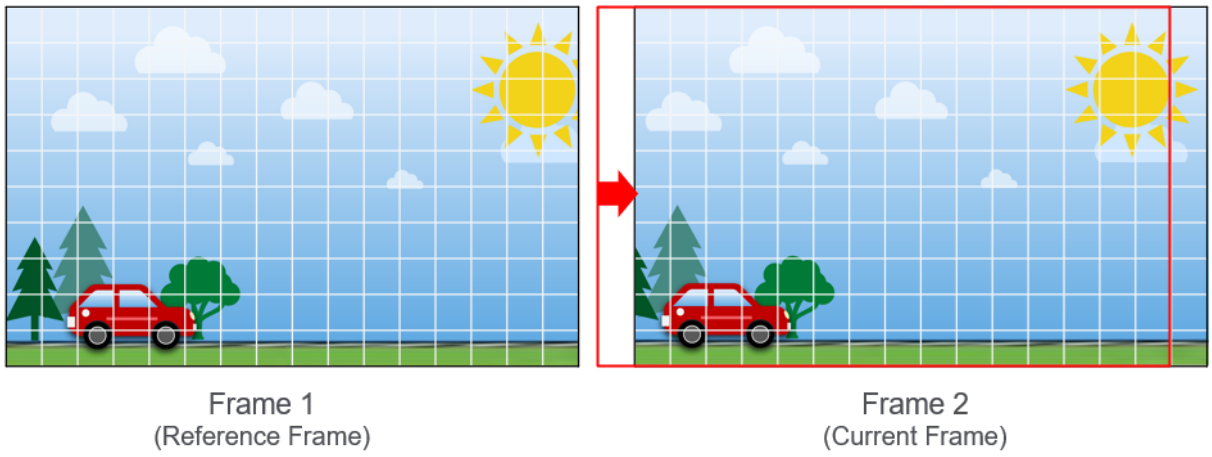
The Background discusses Joint Model Number 1 ("JM1") of the JVT codec. '808Pat., 10:23-12:47. JM1 assigns a "skip mode" to a macroblock, indicating "the macroblock is to be copied from the reference video frame without using motion compensated prediction." '808Pat., 10:63-67. "The motion estimation for the skip mode is a very simple one [because] a constant zero valued vector is assigned for this mode." '808Pat., 11:8-11. For non-skip mode macroblocks, on the other hand, a "cost function" "typically combines the prediction error with number of estimated bits needed to code the macroblock and thus measures the relative efficiency of each coding mode" to decide how the macroblock is divided into subblocks. '808Pat., 11:11-18.

The figure below illustrates the use of skip mode for macroblocks (highlighted yellow), where the macroblocks are copied from the reference frame ("Frame 1") for use in the current frame ("Frame 2") without motion compensated prediction. Saber, ¶49.



The Background also discusses global motion compensation and global motion vector coding. '808Pat., 12:48-14:10. "[T]he idea behind global motion compensation is to generate a reference frame for motion compensation that cancels the effects of global motion," but doing so requires "complex operations." '808Pat., 12:59-66. Global motion vector coding is a "simplified version of global motion compensation" where "[t]he reference frame is used as it is, but additional information is transmitted to describe the global motion and additional macroblock modes are used to indicate when global motion vectors are used." '808Pat., 13:31-36.

The figure below illustrates global motion compensation, which can be used, e.g., to code camera panning, where a current frame ("Frame 2") is predicted based on "global" motion of a reference frame ("Frame 1"). Saber, ¶49.



The ’808 Patent purports to improve upon the prior art through “a redefinition of the skip mode concept used in JM1 of the JVT codec.” ’808Pat., 14:14-18. The ’808 Patent “redefines” the skip mode concept such “that a macroblock assigned to skip mode is either associated with a zero (non-active) motion vector, in which case it is treated in the same way as a conventional skip mode macroblock and copied directly from the reference frame, or it is associated with a non-zero (active) motion vector.” ’808Pat., 14:23-29, 17:45-52, 19:28-51. Non-zero motion vectors are assigned based on the characteristics of the motion of previously coded neighboring macroblocks or sub-blocks. ’808Pat., 14:29-37, 14:52-60, 17:58-64.

According to the ’808 Patent, “minimal…modification is required to incorporate the method according to the invention into existing video coding systems that employ the concept of skip mode macroblocks,” and the claimed encoder and decoder of the ’808 Patent are “substantially identical” to the prior art encoder and decoder. ’808Pat., 15:11-14, 16:7-14, 20:29-32; *compare* Figs. 1-2 (prior art

encoder and decoder, respectively), *with* Figs. 6-7 (encoder and decoder, respectively, of the purported invention) (all reproduced below).

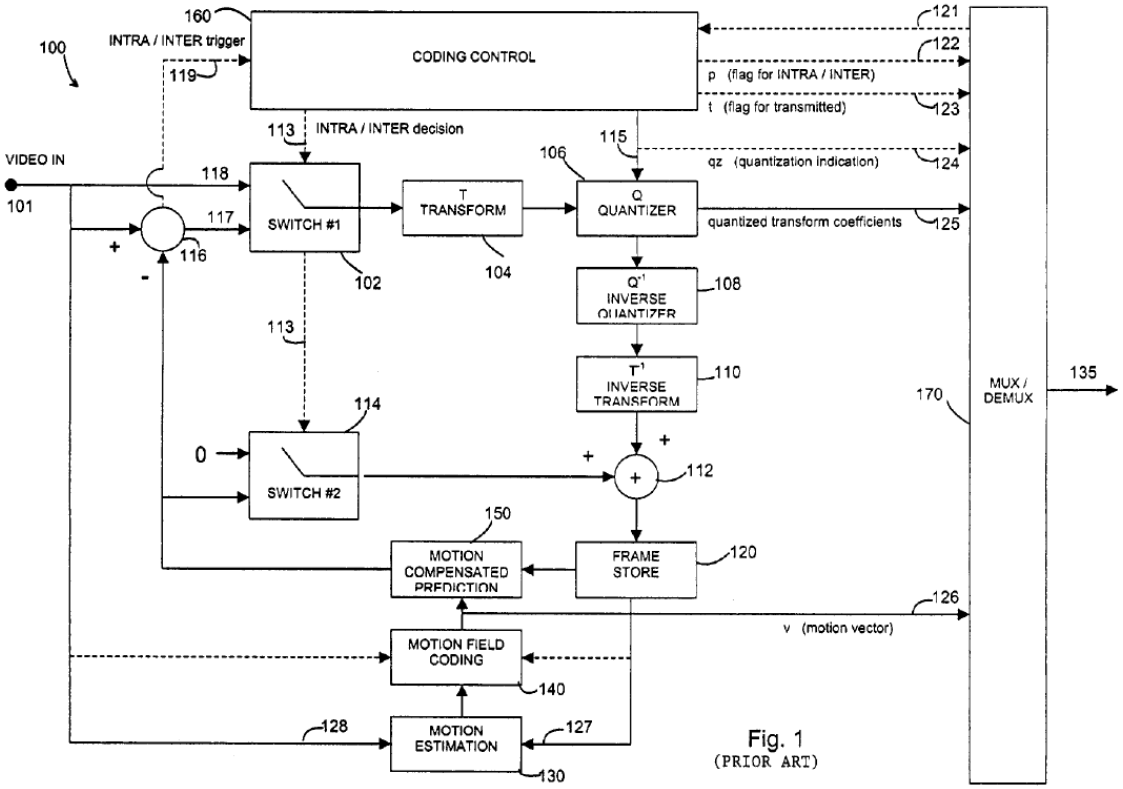


Fig. 1
(PRIOR ART)

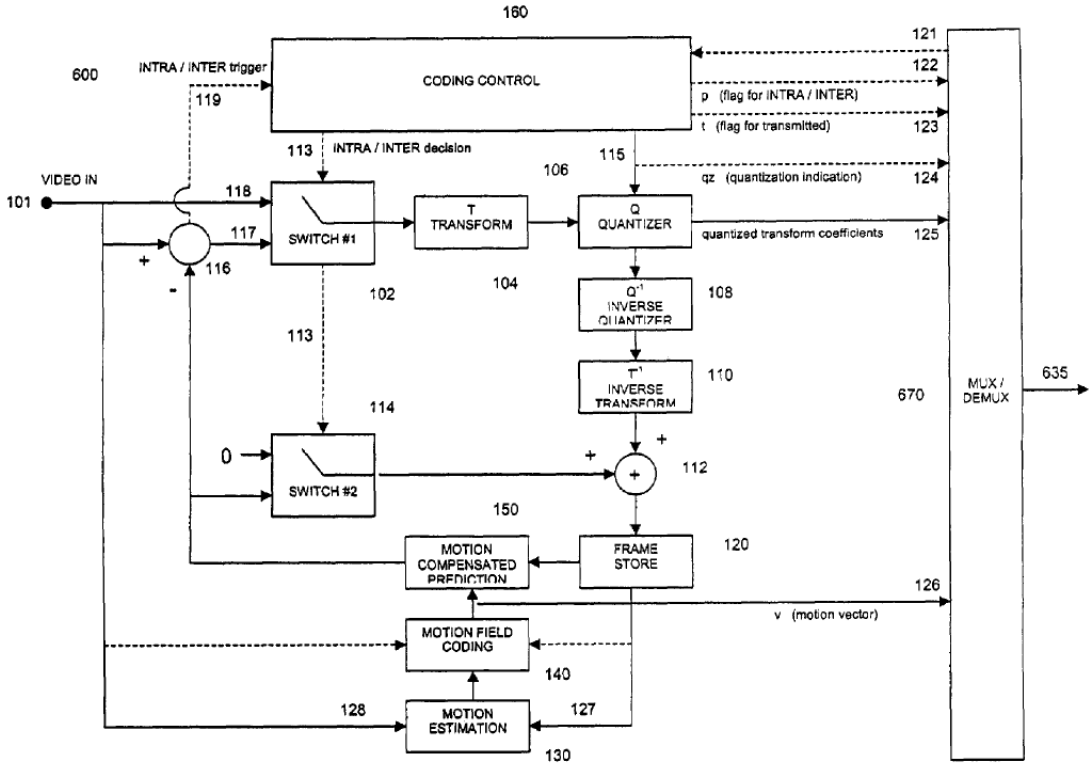


FIG. 6

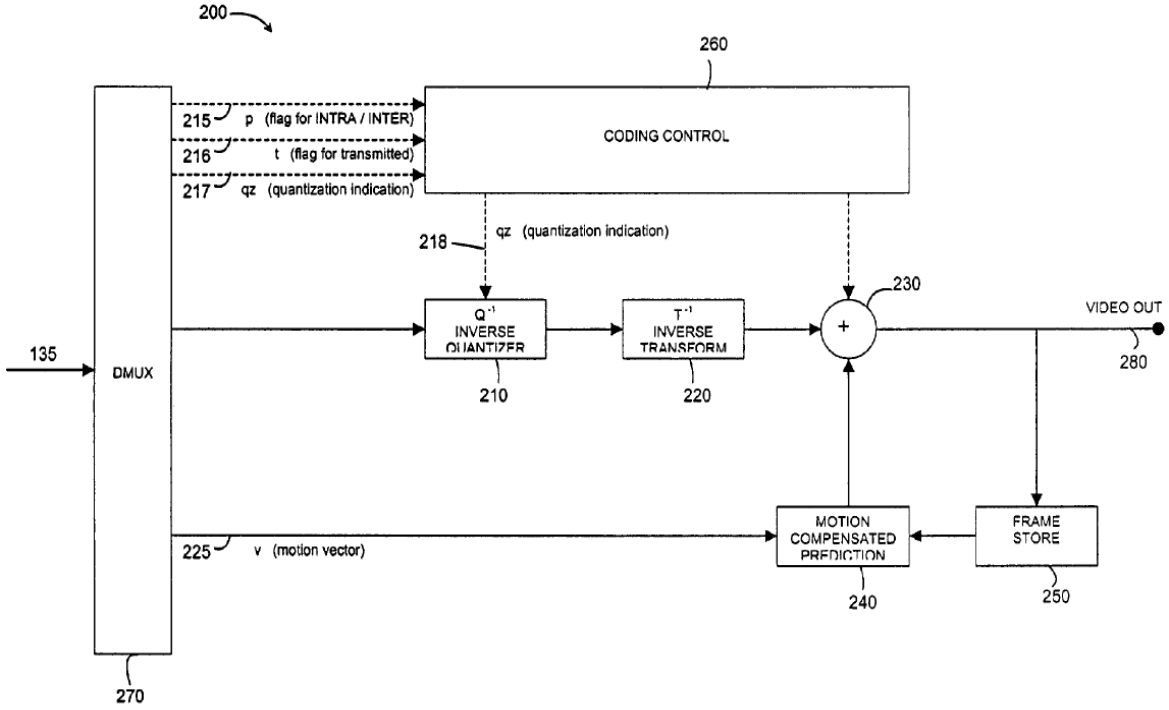


Fig. 2
(PRIOR ART)

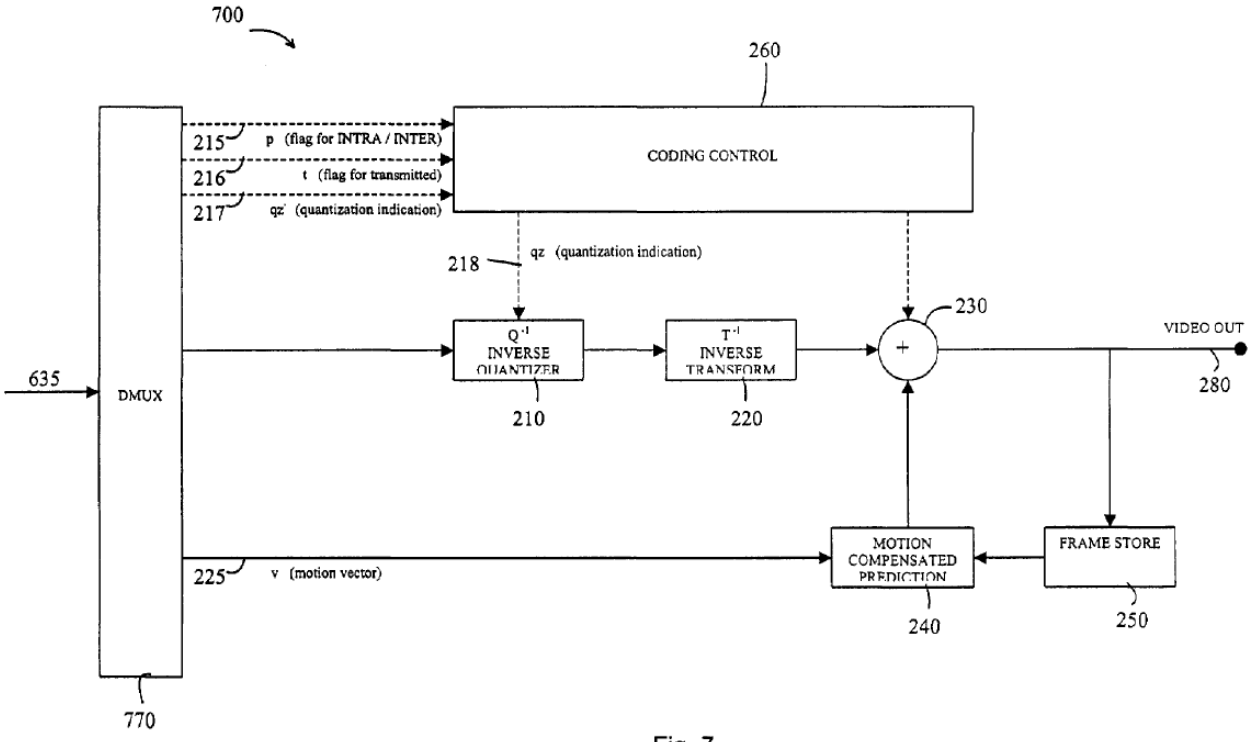


Fig. 7

V. LEVEL OF ORDINARY SKILL

A person of ordinary skill in the art ("POSITA") at the time of the '808 Patent's alleged invention 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. Saber, ¶26.

VI. CLAIM CONSTRUCTION

Claims should be construed "in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the

prosecution history pertaining to the patent." 37 C.F.R. §42.100(b); *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc).

In a previous challenge to the '808 Patent by Amazon ("Amazon IPR"), the Board preliminarily construed the term "skip coding mode" to mean "a coding mode in which a zero (non-active) motion vector or a non-zero (active) motion vector is associated with each skip mode macroblock, depending on the characteristics of the motion in image segments surrounding the macroblock in question" based on lexicography. *Amazon.com, Inc. v. Nokia Techs. Oy*, IPR2024-00847, Paper 11, 18-19 (Dec. 11, 2024); *Amazon.com, Inc. v. Nokia Techs. Oy*, IPR2024-00848, Paper 11, 19 (Dec. 11, 2024); '808pat, 17:45-52.[7] For purposes of this IPR, Petitioner applies the Board's previous preliminary construction, and the construction that PO advocated for, of "skip coding mode."

VII. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Obviousness over Karczewicz

1. Karczewicz

Karczewicz teaches an "apparatus, and an associated method, for encoding, and decoding, a video sequence utilizing motion compensated prediction."

[7] PO agreed with this construction. *See* IPR2024-00847, Paper 21, 15-16; IPR2024-00848, Paper 21, 15-16.

Karczewicz, 1:7-8. Karczewicz teaches five different modes for coding a segment of a frame of a video sequence. Karczewicz, 31:14-25, claims 61("defining a set of coding modes for said neighboring segment…"), 64-65 (describing each of five "coding modes"). In some of these modes, "[m]otion fields of a segment are predicted from adjacent segments of a video frame and by using orthogonal affine motion vector field models." Karczewicz, 1:9-10, *see id.*, 14:2-5. Thus, "motion vector fields are formed with a reduced number of bits while still maintaining a low prediction error." Karczewicz, 1:11-13. Karczewicz explains motion vector fields can be thought of as the representation of "motion vectors actually identify[ing] the origin of pixels in the current frame by comparison with the reference frame," so a POSITA would have understood when Karczewicz refers to coding "motion fields" of a segment, it is referring to coding motion vectors for said segment. Saber, ¶73; Karczewicz, 8:3-10, 20:11-13.

Two of Karczewicz's coding modes exploit the similarity between motion vector fields of neighboring segments (typically macroblocks) by computing a motion vector field of a segment from its previously coded neighboring segment. Karczewicz, 7:15-19, 11:1-8, 31:7-9, 31:14-25, claims 61, 64-65. Karczewicz teaches other modes where a small refinement field is also coded to account for differences between motion vectors of neighboring segments. *Id.* Specifically, after coding a first segment, Karczewicz determines whether it is "'more efficient' to

model the second segment in terms of the affine motion model previously determined for the first segment, rather than according to the newly determined affine motion coefficients for the second segment itself." Karczewicz, 14:27-15:1. Karczewicz explains "in many cases, it is found that after projection of the motion field model from a neighboring segment, very little or even zero refinement information is required." Karczewicz, 15:27-29. "This can result in a significant reduction in the amount of data that must be transmitted from encoder to decoder." Karczewicz, 15:29-16:1. If zero refinement is needed, "the motion vector field of the second segment can be predicted purely on the basis of motion coefficients [from the first segment] already stored in the decoder." Karczewicz, 16:2-3. Notably, Karczewicz also explains the second segment "can also be coded by using a zero motion vector ("ZMV") field, e.g., a copy from the reference frame." Karczewicz, 31:20-24.

2. Claim 1 and Dependents

a. Claim 1

(i) 1[pre]

If the preamble is limiting, Karczewicz discloses or suggests 1[pre]. Karczewicz, 1:7-8 ("method, for encoding, and decoding, a video sequence utilizing motion compensated prediction").

(ii) 1[a]

Karczewicz discloses or suggests 1[a]. Saber, ¶¶84-100.

Karczewicz says a "prediction field" is "the motion field model predicted from the [previously coded] neighboring segment(s)," and a "refinement field" is "the difference between the prediction field and the motion field model determined specifically for the segment itself." Karczewicz, 16:18-23. Karczewicz teaches "[t]he sum of the prediction field and the refinement field should thus be equivalent to the motion field model determined by applying the affine motion model to the segment itself." Karczewicz, 16:23-27.

Karczewicz further explains "there are several different ways in which a given segment can be coded" (Karczewicz, 17:1-2), which Karczewicz characterizes as—and a POSITA would have understood to be—different coding modes[8]. Karczewicz, 31:14-25, claims 61, 64-65; Saber, ¶¶85-90.

[8] In the Amazon IPR, PO argued that Karczewicz "does not disclose coding modes." IPR2024-00847, Paper 21, 37-38; *id*., Paper 10, 33. But Karczewicz explicitly teaches "coding modes" (e.g., in claims 61, and 64-65), and the Board relied on such teachings to reject PO's argument and institute the Amazon IPR. *Id.*, Paper 11, 32. PO's additional argument that Karczewicz's coding modes do not "align with the '808 Patent's singular 'skip coding mode'" attempts to import

Karczewicz determines which mode to assign to a segment based on which is the most efficient. Specifically, Karczewicz teaches that when encoding a second segment of an inter frame (here, mapped to the claimed *first segment*), an encoder "considers whether it is 'more efficient' to model the second segment in terms of the affine motion model previously determined for [a] first segment, rather than according to the newly determined affine motion coefficients for the second segment itself." Karczewicz, 14:27-15:1. Similarly, Karczewicz discloses that "the entire refinement field may be set to zero, if that is efficient." Karczewicz, 19:2-5. "Efficiency" is based on "rate distortion," which considers "both a measure of image distortion and a measure of the amount of data required to achieve that level of distortion." Karczewicz, 14:12-17. In other words, Karczewicz balances the reduction in data coding against the potential quality loss of the coded video sequence. Saber, ¶91.

In the first mode, "[t]he segment may be represented as a prediction field only." Karczewicz, 17:9-10, 31:16-17, claim 64. "This situation may arise when

---

limitations such as requiring a "singular" *skip coding mode* or "defining comprehensive coding modes." IPR2024-00847, Paper 21, 37-38. Karczewicz discloses or suggests a *skip coding mode* as actually claimed for the reasons explained herein.

the segment can be adequately represented in terms of the motion field of one or more of its previously coded neighbors and no refinement information is necessary, or in a case where the encoder has found it efficient to reduce the refinement field to zero." Karczewicz, 17:10-14. In other words, Karczewicz teaches coding a second segment based on a neighboring segment (or segments) when efficient. Karczewicz, 14:12-17; Saber, ¶86.

In the second mode, "[t]he segment in question may be coded using a motion model determined specifically for the segment using the reference frame…the prediction field is set to zero and the refinement field is set equal to the motion field model determined from the reference frame." Karczewicz, 17:14-18, 31:20-21, claim 64.

In the third mode, "[a] given segment can be represented as a sum of a prediction field and a refinement field." Karczewicz, 17:8-9, 31:17-20, claim 64.

In the fourth mode, "[t]he segment can also be coded by using a [ZMV] field, e.g., a copy from the reference frame." Karczewicz, 31:22-24, claim 65.

Finally, in the fifth mode, "the segment can be coded using intra coding in which no motion vector field is utilized." Karczewicz, 31:24-25, claim 65.

Karczewicz thus assigns a *skip coding mode* to a second segment of an inter frame[9] (*first segment of a first frame of the sequence*) in instances where the second segment is either encoded using ZMV (Karczewicz, 31:22-24) or with just a prediction field—i.e., a non-zero motion vector associated with a neighboring segment. Karczewicz, 17:8-14, 31:22-24; Saber, ¶92. While Karczewicz lists these two functionalities as separate coding modes, a POSITA would have understood the functionality of these two modes together corresponds to the '808 Patent's "modified" "skip coding mode," which groups these two independent functionalities and "redefines" them as the "skip coding mode." Saber, ¶¶92-93; '808Pat., 17:54-18:11. Karczewicz teaches a single indicator, the MCI, which has a zero value only in the two modes comprising the functionality of claim 1's *skip coding mode* discussed above. Thus, a POSITA would have understood a zero MCI bit indicates that a segment is assigned a "skip coding mode." Saber, ¶¶92-93.

Specifically, "the presence of motion coefficients in the bitstream is signaled by one bit whenever refinement or nonrefinement motion coefficients can be expected. This bit is referred to as...(MCI)." Karczewicz, 37:5-8. "An all-zero

[9] Karczewicz transmits a sequence's first frame in INTRA format, and a POSITA would have understood that the recited "first frame in the sequence" refers to the sequence's first inter-coded frame. Karczewicz, 13:17-23; Saber, ¶100.

pattern [of motion coefficients]…can be signaled by the MCI bit alone." *Id.*, 37:12-13. Thus, a zero MCI bit indicates Karczewicz's two inter-coding modes corresponding to claim 1's "redefined" *skip coding mode*. Specifically, when a segment is coded with ZMV, the MCI bit is zero because that indicates "[a]n all-zero pattern [of motion coefficients]." *Id*. And when a segment is coded with just a prediction field, the MCI bit is also zero because it indicates that there are no motion coefficients for the segment (i.e., no refinement field). Karczewicz, 37:5-8. These are the only two modes taught by Karczewicz that include a zero MCI bit, and their functionality corresponds to the *skip coding mode* in the '808 Patent. Accordingly, a POSITA would have understood that if a segment is coded with a zero MCI bit, it is assigned a singular *skip coding mode*. Saber, ¶93.

To the extent PO argues that a POSITA would not have understood coding a segment with either a zero motion vector or a prediction field to be a single "mode" (as indicated by an MCI bit set to 0), it would have been obvious to a POSITA to combine these teachings into a single coding mode. Saber, ¶¶94-95. Doing so would reduce complexity and improve coding efficiency by reducing the number of available coding modes available to the encoder and decoder while accounting for any possible redundancy between Karczewicz's first and fourth coding modes (e.g., listing as two separate modes the same functionality that a single mode accomplishes), without removing any of Karczewicz's functionality. *Id.* Indeed,

Karczewicz already teaches how to signal the combined functionality of these two modes (setting the MCI bit to zero), so a POSITA would not have needed to make any modifications to Karczewicz beyond redefining the mode and would therefore have had a reasonable expectation of success[10]. *Id.*

Finally, a POSITA would have understood the decision to code a second segment with ZMV or predicted non-zero motion vector ("PNZMV") depends on the motion characteristics in the surrounding segment(s), consistent with the meaning of *skip coding mode*, because Karczewicz determines the efficient coding mode by balancing distortion due to coding the second segment with reference to a neighboring segment against the amount of data required to transmit the second segment's motion vector information. Karczewicz, 15:15-20; Saber, ¶96. In other

[10] In the Amazon IPR, PO argued that Karczewicz's modes for coding a segment with just a prediction field and with a zero motion vector are "mutually exclusive." IPR2024-00847, Paper 21, 44-47. That is incorrect. After an MCI bit set to zero indicates a *skip coding mode*, Karczewicz teaches another bit signaling whether a prediction is used, and if necessary, which surrounding segments should be used for the prediction field, as explained in §VII.A.2.a(v). Saber, ¶95. This additional signaling would have motivated a POSITA to combine the two coding modes with a reasonable expectation of success. *Id*.

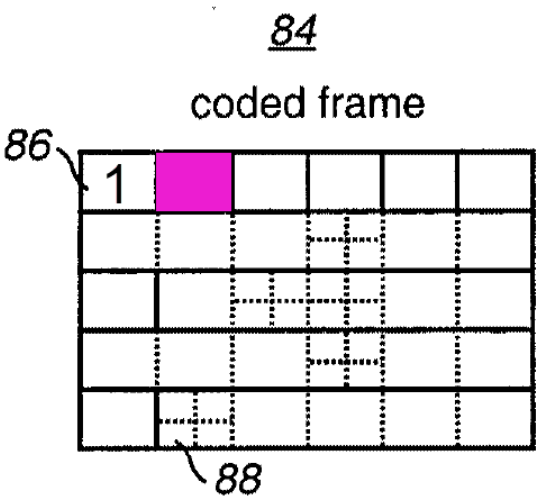
words, Karczewicz considers the *characteristics of motion in image segments surrounding the macroblock in question* compared to the motion characteristics of the macroblock in question when determining an efficient way for coding the macroblock in question. Saber, ¶96.

Furthermore, if the encoder decides that it is efficient to code a macroblock based on the motion field(s) of one or more previously coded neighbors, the *non-zero (active) motion vector [] associated with [the] skip mode macroblock depend[s] on the characteristics of the motion in image segments surrounding the macroblock in question*. Saber, ¶97. Specifically, Karczewicz teaches that the motion vector field for a macroblock can be "obtained from a combination of affine motion vector fields of several neighboring segments using some particular method which is known both to the encoder and to the decoder. Such method is, for example, averaging or determining the median of horizontal and vertical motion vector field components." Karczewicz, 25:20-24. These methods can be applied based on one, two, or four neighboring segments. Karczewicz, 16:4-10 (explaining a situation in which "the second segment has only one neighbor that can be used to predict its motion"), 16:10-20 ("segments in subsequent rows have [two] neighbors[,] to the left and above" that the encoder can use to predict the motion coefficients for a macroblock, "using the motion field model for both neighbors."), 18:17-28 ("In this case, a

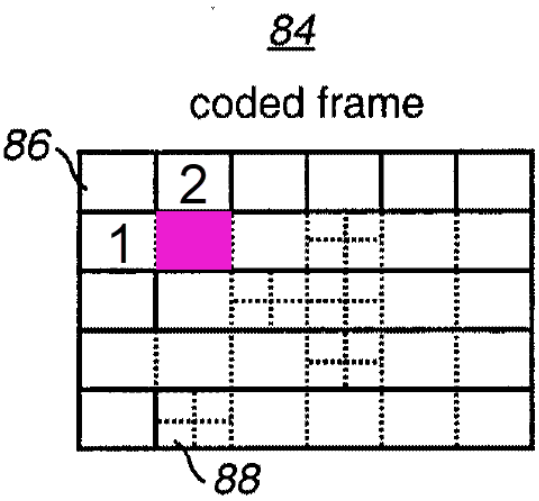
general 16x16 pixel segment has four immediately neighboring 8x8 pixel sub-segments that may be considered").

Karczewicz's Figure 4 illustrates how a video frame may be divided into segments, Karczewicz, 31:3-9, and has been annotated to show how Karczewicz teaches considering one, two, or four surrounding segments for the motion field of a currently coded macroblock (pink):

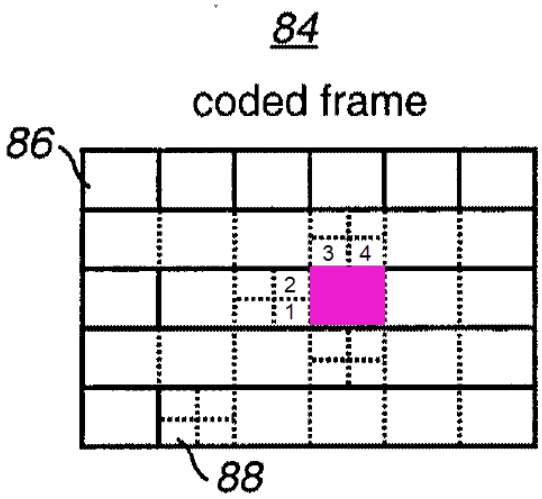
One Segment



Two Segments



Four Segments:



A POSITA would have understood that Karczewicz's description of using the motion field model of one, two, or four "neighboring" segments corresponds to *motion in image segments surrounding the macroblock in question* because both Karczewicz and the '808 Patent use previously coded segments to the left and above the macroblock being coded. Saber, ¶97.

In the Amazon IPR, PO argued that Karczewicz does not disclose a *skip coding mode* "that considers surrounding motion to determine whether to treat the skip coding mode as active (and then generate a predicted motion vector) or non-active (and then signal a zero-motion vector) to best predict a given image block from a reference frame." IPR2024-00847, Paper 21, 40. But that requirement is not present in the claim language or the agreed construction of *skip coding mode*, as detailed in §VII.B.4.a(ii). Nevertheless, even applying PO's interpretation of the

claim, Karczewicz still discloses a *skip coding mode*. As explained above, Karczewicz considers the most efficient way to code a segment based on the motion of surrounding segments and the actual motion of the segment being coded. Karczewicz, 14:27-15:6, 19:2-5. Based on an efficiency determination, Karczewicz discloses that a segment may be coded using a prediction field alone (Karczewicz, 15:26-16:3), which PO characterized as "determine[ing] to treat the skip coding mode as active." Saber, ¶¶98-99. Or if it is efficient to do so, Karczewicz teaches setting the entire refinement field to zero. (Karczewicz 19:2-9.) A POSITA would have understood that Karczewicz applies this analysis even when a segment is not coded with a prediction field (Karczewicz, 14:12-17) such that setting a refinement field to zero, if efficient, would result in the segment being coded with a zero motion vector, which PO characterized as "determine[ing] to treat the skip coding mode as…non-active (and then signal a zero-motion vector).[11] Saber, ¶99.

[11] The construction for *skip coding mode* is silent regarding whether prediction error is or can be sent. To the extent PO argues that a *skip coding mode* additionally requires that no residual information is provided for a skipped macroblock, Karczewicz discloses or suggests this scenario for the reasons discussed in §VII.A.2.e.

(iii) 1[b]

Karczewicz discloses or suggests 1[b]. Saber, ¶¶101-102.

Karczewicz assigns a *skip coding mode* to a "second segment" (which corresponds to the claimed *first segment*) based on an efficiency determination. §VII.A.2.a(ii). The segment is assigned either a prediction motion vector field "extrapolated from a neighboring segment" or a "[ZMV] field, e.g., a copy from the reference frame." Karczewicz, 31:16-24.

Karczewicz further teaches that the *assigning* is *based at least in part on the motion information of a second segment neighboring the first segment*. If more efficient, a second segment (*first segment*) is coded with reference to the motion information of a previously coded segment or segments (*second segment*). Karczewicz, 15:15-20. Accordingly, a POSITA would have understood the *assigning* of ZMV or PNZMV for the skip coding mode of the second segment (*first segment*) is based at least in part on the motion information of a neighboring segment (*second segment*) because Karczewicz's efficiency determination weighs using motion coefficients from a neighboring, previously coded segment, against motion coefficients calculated for the second segment.[12] *Id.*; Saber, ¶101.

[12] One way Karczewicz determines the most efficient mode is by minimizing the value λ in its equations 7-10. Karczewicz 24:18-27:11.

Karczewicz's PNZMV is also *based at least in part on the motion information of a second segment neighboring the first segment*. Karczewicz's prediction motion vector field is "extrapolated from a neighboring segment" or segments. Karczewicz, 31:16-17, 25:14-17, 25:19-23. Thus, the prediction motion vector field of Karczewicz's *skip coding mode* (*[PNZMV] for the skip coding mode*) is based at least in part on the motion information of a previously coded neighboring segment (*second segment neighboring the first segment*). Saber, ¶102.

(iv) 1[c]

Karczewicz teaches *forming a prediction* for its second segment (*first segment*) with respect to a reference frame by assigning to the second segment either ZMV or a motion vector based on one or more previously coded segments (*second segment*). Karczewicz, 21:16-21 ("the compressed motion vectors and the reference frame are used to construct a prediction frame[]. The prediction frame is a coded version of the current frame[], calculated using the motion vector field…and the pixel values of the reference frame."), 16:16-18 ("the encoder can try to predict the motion coefficients for a given segment using the motion field model for the segment above it or to the left"), 20:17-20, 21:6-21, 25:13-24, 26:13-20, 27:15-30, 31:13-24.

(v) 1[d]

Karczewicz discloses or suggests 1[d]. Saber, ¶¶104-106.

Karczewicz teaches a zero MCI is coded only with a segment that is assigned a *skip coding mode*. §VII.A.2.a(ii). Thus, a POSITA would have understood coding a zero MCI bit for a segment to be *providing in an encoded bitstream an indication of the skip coding mode*. Saber, ¶104.

Karczewicz further discloses "one bit is sent to the decoder 14 to signal whether the prediction field of a neighbor is used or not, but only in the case when there is at least one prediction neighbor candidate." Karczewicz, 32:18-20, 17:23-28 ("Whenever a segment is coded using a prediction field…it is necessary to provide information about the neighboring segment(s) used in the prediction. It is not necessary to transmit any motion coefficient data because the motion field model(s) of the previously coded neighboring segment(s) are already known to the (i.e., stored in) the decoder."). Karczewicz thus teaches that *no further motion vector information for the first segment is coded in the encoded bitstream*. Saber, ¶105. The additional information related to the prediction field (the bit indicating use of a prediction field or information identifying surrounding segments used to calculate the prediction field) is not *motion vector information*.[13] *Id*.

[13] This limitation requires only that *no further motion vector information for the first segment is coded*, which a POSITA would have understood not to limit

In the Amazon IPR, PO argued that "Karczewicz teaches away from the '808 Patent by explicitly signaling a motion model and a motion coefficient indication." IPR2024-00847, Paper 21, 41-42. Tellingly, PO does not tie these arguments to specific claim language. *See id.* To the extent that PO's argument regarding "signaling a motion model" is meant to relate to this limitation's recitation that *no further motion vector information for the first segment is coded*, Karczewicz discloses this limitation as explained above. Similarly, to the extent PO's argument regarding the "signaling…a motion coefficient indication" relates to the MCI bit, the MCI bit is *an indication of the skip coding mode* as explained herein.

b. Claim 2

Karczewicz discloses or suggests claim 2's additional limitation. As discussed for claim 1, the prediction field for Karczewicz's second segment (*first segment*) is "represented in terms of the motion field of one or more of its previously

---

whether a residual is coded because residuals contain pixel values rather than motion vector information. Saber, ¶106. To the extent PO meant that 1[d] additionally requires that no residual information is provided for a skipped macroblock, Karczewicz discloses or suggests this scenario for the reasons discussed in §VII.A.2.e.

coded neighbors" (*second segment*). Karczewicz, 17:10-14, 14:27-15:6, 16:4-30; 17:23-28, 31:8-9.

c. Claim 6

Karczewicz discloses or suggests claim 6's additional limitation. Saber, ¶¶108-110. As explained in §§VII.A.2.a(ii), VII.A.2.a(iii), Karczewicz codes a second segment (*first segment*) with reference to a previously coded neighboring segment (*second segment*) via a prediction motion vector field or with ZMV. Karczewicz, 31:22-24. A POSITA would have understood when a previously coded neighboring segment has ZMV, Karczewicz codes its second segment with ZMV if it is efficient. Saber, ¶94. In such a case, when ZMV is coded, the segment is copied from the reference frame—i.e., *formed with respect to a corresponding segment of the reference frame associated with the [ZMV]*. Karczewicz, 31:22-24; Saber, ¶108.

To the extent PO argues that Karczewicz does not allow neighboring segments with ZMVs to be candidates for a prediction motion vector field (based on, *e.g.*, Karczewicz, 32:20-22), PO's argument would be misplaced because Karczewicz renders this claim obvious regardless. Saber, ¶¶109-110. This disclosure in Karczewicz is specifically related to the ***decoder*** side, and it is harmonized with Karczewicz's encoder disclosures, 32:18-20. Specifically, if a second segment is encoded with ZMV, which may be based on the motion vectors of its neighbors if it is efficient to do so in the rate-distortion sense, then the decoder would not receive

a prediction motion vector field for that segment (because it is coded with ZMV), and there would not be any prediction neighbor candidates. Saber, ¶110. At the decoder, there are prediction motion vector field candidates and coefficients only if a segment is coded with a non-ZMV. *Id*. But this does not affect Karczewicz's teachings about how a segment may be encoded with ZMV. *Id*.

d. Claim 23

(i) 23[pre]

If the preamble is limiting, Karczewicz discloses or suggests a previously coded neighboring segment (*second segment*) may have *a non-zero motion vector*. Karczewicz, 32:20-22.

(ii) 23[a]

Karczewicz discloses or suggests 23[a] for the same reasons as discussed in §§VII.A.2.a(ii), VII.A.2.a(iii). Karczewicz's second segment "can be coded using only prediction motion vector fields extrapolated from a neighboring segment." Karczewicz, 31:16-20. Specifically, "[o]nce the neighboring segment has been selected for the prediction of the current segment, the prediction motion vector field is simply the extrapolation of the motion vector field of the segment inside the pixel domain covered by the current segment as follows:

EQUATION 20

$$\Delta x_{prd}(x,y) = \beta_1^k + \beta_2^k \cdot (y - y_o^k) + \beta_3^k \cdot (x - x_o^k)$$

$$\Delta y_{prd}(x,y) = \beta_4^k + \beta_5^k \cdot (y - y_o^k) + \beta_6^k \cdot (x - x_o^k)$$

where x_0^k, y_0^k are coordinates of the upper left-most pixel in the neighboring segment S_k and β_1^k,...β_6^k are integer-valued coefficients corresponding to the motion field of segment S_k. In equation 20, the superscript ^{k} indicates that the coefficients are associated with the neighboring segment S_k." Karczewicz 33:18-34:2. Where the one or several neighboring segments have a non-zero motion vector field, a POSITA would have understood Karczewicz *deriv[es] the [PNZMV] based at least in part on the motion vector of the second segmen*t. Saber, ¶¶112-113.

(iii) 23[b]

Karczewicz discloses or suggests 23[b]. As discussed in §VII.A.2.a(iii), after the *skip coding mode* is assigned, Karczewicz's second segment (*first segment*) is coded with a prediction motion vector field "extrapolated from a neighboring segment." Karczewicz, 31:16-17. Karczewicz thus codes (*assigns*) its second segment with a prediction motion vector field extrapolated from a neighboring segment (*[PNZMV]*).

(iv) 23[c]

Karczewicz discloses or suggests 23[c]. As explained in §VII.A.2.a(iv), Karczewicz forms a motion vector prediction field for its second segment (*first segment*) with respect to a reference frame based on one or more previously-coded segments (*second segment*). Karczewicz, 13:29-14:2, 16:16-18, 21:6-21, 25:13-24, 26:13-20, 31:13-24. Where the *skip coding mode* is assigned solely to a prediction motion vector field (Karczewicz, 31:16-17), a POSITA would have understood *the prediction for the first segment is formed by motion compensated prediction with respect to the reference frame based at least in part on the [PNZMV].*

a. Claim 24

Karczewicz discloses or suggests claim 24's additional limitation. Karczewicz does not discuss how a previously coded neighboring segment (*second segment*) is predicted because Karczewicz only requires that the neighboring segment be previously coded (and thus known to both the encoder and decoder). Saber, ¶116; Karczewicz 17:25-28 ("motion field model(s) of the previously coded neighboring segment(s) are already known to (i.e., stored in) the decoder."). Karczewicz discloses, however, that a segment assigned ZMV is copied from the reference frame. Karczewicz, 13:29-30, 31:22-24. Therefore, if a previously coded neighboring segment (*second segment*) has ZMV, it is predicted using the reference picture. When the previously coded neighboring segment has ZMV, Karczewicz

may decide that it is efficient for its second segment (*first segment*) to have ZMV. §VII.A.2.a(ii). A segment coded with ZMV is copied from the reference frame—i.e., *formed with respect to a corresponding segment of the reference frame associated with the [ZMV].*

b. Claim 25

Karczewicz discloses or suggests claim 25's additional limitation for the same reasons discussed in §VII.A.2.a because Karczewicz's reference frame may immediately precede the current frame. Karczewicz, 5:30-6:2.

c. Claim 28

Karczewicz discloses or suggest claim 28's additional limitation for the same reasons discussed in §VII.A.2.a(iii). Karczewicz derives a motion vector for a second segment based upon the motion field of one or more of its previously coded neighbors. Karczewicz, 17:8-14.

d. Claim 29

Karczewicz discloses or suggests claim 29's additional limitation for the same reasons discussed in §VII.A.2.a(iii). Karczewicz discloses the motion vector for the second segment (*first segment*) may be derived from "one or more of its previously coded neighbors." Karczewicz, 17:10-14, 16:16-19. Accordingly, Karczewicz discloses *deriving the PNZMV* for its second segment from at least two previously coded neighboring segments (*second segment* and *third segment*).

e. Claim 30

Karczewicz discloses or suggests claim 30's additional limitation for the reasons discussed in §§VII.A.2.a(ii), VII.A.2.a(iii). Karczewicz's prediction error is calculated as "the differences in pixel value between the current input video frame and the MC predicted version of the frame." Karczewicz, 20:22-25. A POSITA would have understood that in some instances there would be no difference between the pixel values for the motion-compensated predicted version of the frame and the actual version of the frame for the recited *first segment*. Saber, ¶120. In such a case, the prediction error (*residual*) would be zero. Saber, ¶121.

Karczewicz does not specify whether or how a prediction error of zero would be encoded. But it would have been obvious to a POSITA not to encode a zero prediction error based on the teachings of Karczewicz that the MCI bit can be used to indicate an all-zero field rather than encoding an all-zero motion field. Karczewicz, 37:5-13; Saber, ¶121. A POSITA would have been motivated to apply the same principle when an all-zero prediction error is calculated, using a different bit to indicate an all-zero prediction error because it would have provided a more efficient way of coding an all-zero prediction error. *Id*. It also would have amounted to no more than using a known technique (indicating an all-zero field with a single bit) to improve a similar method (sending an all-zero prediction error) in the same way (efficiently indicating all zero values). It also would have been obvious to use

another bit or indicator—similar to the MCI bit—to code zero prediction error because it would have been one of a limited number of choices (e.g., coding zero prediction error or indicating zero prediction error without coding it). *Id.*

In addition, a POSITA would have been motivated not to encode a zero prediction error based on established coding standards at the time of the '808 Patent, including H.262, H.263, and H.26L. Saber, ¶122. In those standards, prediction error is transformed, scanned in a zig-zag order, and run-length coded, meaning that each run of consecutive zero values is paired with an encoding of a non-zero value terminating the run. Saber 122 (citing Karczewicz, 24:12-14); Wang, 294, Fig. 9.12; Bhaskaran, 168; H.263 (EX-1009) at Section 5.4.2, Table 16, Table 17, pp. 41-44, Section 6.2, pp. 46-47; JM-1r1 (EX-1012) at Section 4.3, pp. 30-32. A special flag called LAST or End of Block (EOB) indicates that all remaining values in the run-length coding are zero. H.263 at Section 5.4.2 on p. 41; JM-1r1 (EX-1012) at 32. Based on these teachings, a POSITA would have understood that if zero prediction error is calculated, it would not be encoded. A POSITA would have been motivated to communicate a zero prediction error more efficiently (e.g., by a bit indicating an all-zero prediction error or by using a LAST or EOB flag to code the prediction error). Saber, ¶122.

3. Claim 7 and Dependents

a. Claim 7

(i) 7[pre]

If the preamble is limiting, Karczewicz discloses or suggests it. Karczewicz, 1:6-8 ("the present invention relates to…an associated method, for encoding, and decoding, a video sequence utilizing motion compensated prediction").

(ii) 7[a]

Karczewicz discloses or suggests 7[a]. As explained in §§VII.A.2.a(ii), VII.A.2.a(v), Karczewicz teaches sending to the decoder (and the decoder receiving) a zero MCI bit to signal that no motion coefficients are sent for a segment (Karczewicz, 37:12-13), and a POSITA would have understood a zero MCI bit to be *an indication of a skip coding mode*. Saber, ¶¶104, 124.

(iii) 7[b]

Karczewicz discloses or suggests 7[b] for the same reasons discussed in §VII.A.2.a(iii) because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment based on the received prediction and refinement fields for the segment. Saber, ¶125; Karczewicz, 27:17-22. This includes assigning the motion vector information received from the encoder, which in the *skip coding mode* is either a *zero motion vector* or a prediction field that indicates a *predicted nonzero motion vector* calculated from the motion

vectors of previously coded surrounding segments indicated by the encoder. Saber, ¶125.

In the Amazon IPR, PO argued that Karczewicz "does not [teach] the challenged decoding claims" because Karczewicz's decoder does not perform a rate distortion analysis. IPR2024-00847, Paper 21, 47-49. PO did not tie this argument to claim language other than *skip coding mode*. *Id.*, 47-49. To the extent PO meant that Karczewicz does not disclose the *assigning* of 7[b] because the decoder does not perform an efficiency determination like the encoder, that is incorrect. As explained in §VII.A.2.a(ii), Karczewicz makes an efficiency determination that results in the encoder *assigning a skip coding mode.* Unlike the encoding claims, the decoding claims require only *receiving an indication of a skip coding mode*, which Karczewicz teaches for the reasons discussed in §VII.A.3.a(ii). The *assigning…a zero motion vector* of 7[b] is performed at the decoder if, after receiving a zero MCI bit, the decoder does not receive a bit indicating that a prediction field is used. Saber, ¶125. And the *assigning…a predicted non-zero motion vector* of 7[b] is performed at the decoder if it receives a bit indicating a prediction motion vector field is used. Karczewicz, 32:18-20. In either situation, the decision of whether to code a segment based on a prediction field (and thus send a bit indicating the use of a prediction field) is provided by the encoder *based at least in part on the motion information of a second segment*, as explained in §VII.A.2.a(ii). A POSITA would have therefore

understood that the *assigning* of 7[b] is also *based at least in part on the motion information of a second segment neighboring the first segment* because the motion information is used by the encoder to determine whether to code the segment with a prediction field (and thus instruct the decoder to decode the segment using a prediction field). Saber, ¶127. And for the reasons explained in §VII.A.2.a(ii), the *predicted non-zero motion vector* is *based at least in part on the motion of a second segment neighboring the first segment*.

(iv) 7[c]

Karczewicz discloses or suggests 7[c] for the same reasons discussed in §VII.A.2.a(iv) because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

b. Claim 9

Karczewicz discloses or suggests claim 9's additional limitation for the same reasons discussed in §VII.A.2.c because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction for its second segment (*first segment*) based on a previously coded, neighboring, first segment (*second segment*). Karczewicz, 27:17-22. A POSITA would have understood the previously decoded neighboring first segment to be *a segment in a previously decoded region surrounding the first segment*. Saber, ¶129.

c. Claim 32

Karczewicz discloses or suggests claim 32's additional limitation for the same reasons discussed in §VII.A.2.c because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

d. Claim 34

Karczewicz discloses or suggests claim 34's additional limitation for the same reasons discussed in §VII.A.2.d because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

e. Claim 36

Karczewicz discloses or suggests claim 36's additional limitation for the same reasons discussed in §VII.A.2.a because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

f. Claim 39

Karczewicz discloses or suggests claim 39's additional limitation for the same reasons discussed in §VII.A.2.c because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

g. Claim 40

Karczewicz discloses or suggests claim 40's additional limitation for the same reasons discussed in §VII.A.2.d because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

h. Claim 41

Karczewicz discloses or suggests claim 41's additional limitation for the same reasons discussed in §VII.A.2.e because Karczewicz's decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

i. Claim 43

Karczewicz discloses or suggests claim 43's additional limitation for the same reasons discussed in §VII.A.2.b because the decoding method involves the same steps as the encoding method for forming a prediction of a segment. Karczewicz, 27:17-22.

j. Claim 65

Karczewicz discloses or suggests claim 65's additional limitation for the same reasons discussed in §VII.A.2.a(v) because Karczewicz's "*skip coding mode*" does not provide a refinement motion vector field (i.e., *further motion vector information for the first segment*).

B. Ground 2: Obviousness over MPEG-1 and H.263

1. MPEG-1

ISO/IEC 11172-2:1993, Information technology — Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s — Part 2: Video ("MPEG-1") was published on August 1, 1993.

MPEG-1 discloses three picture types relevant to this Petition: I-Pictures, P-Pictures, and B-Pictures. MPEG-1, §0.2.1. I-Pictures are intra-coded and "are coded without reference to other pictures." *Id.* P-Pictures "are coded…using motion compensated prediction from a past intra or predictive coded picture." *Id.* B-Pictures "require both past and future reference pictures for motion compensation." *Id.* The relationship between these three picture types is illustrated below:

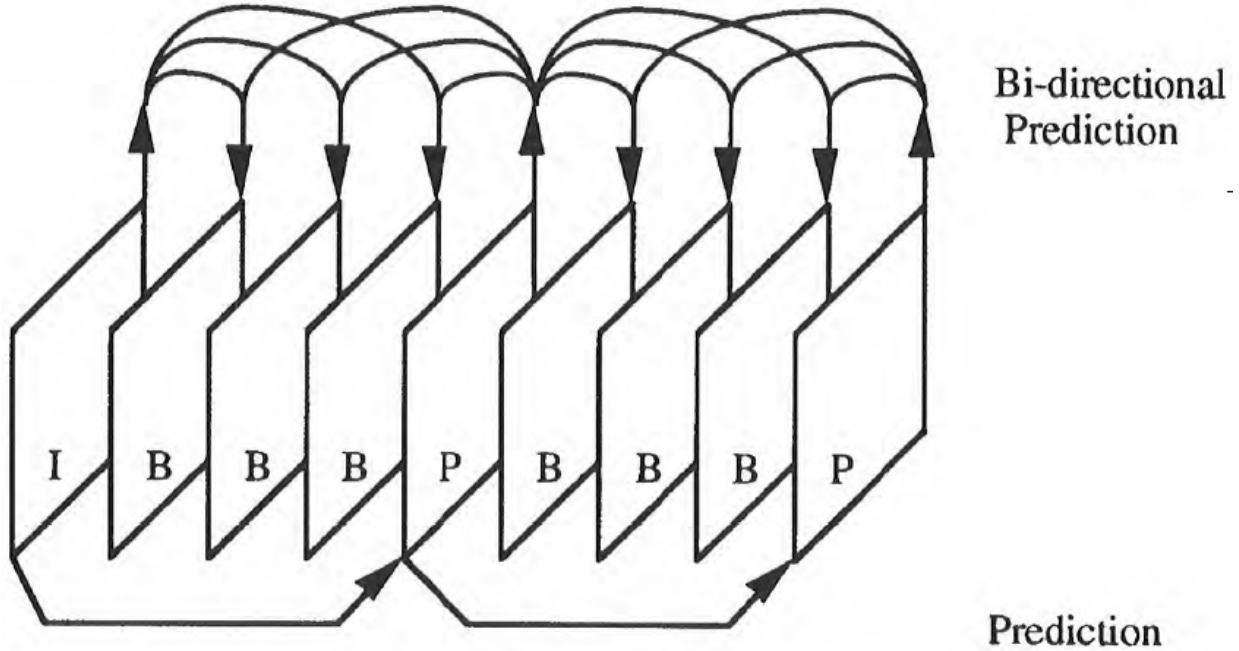
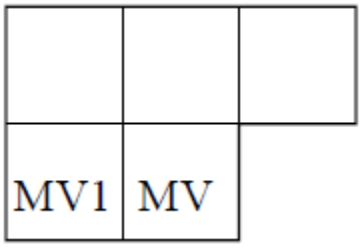


Figure 1 -- Example of temporal picture structure

MPEG-1, Figure 1.

One technique used by MPEG-1 for coding P and B Pictures is "skipped macroblocks," for which no data are stored or coded. MPEG-1, §§2.1.131, 2.4.4.4 ("For some macroblocks there are no coded data, that is neither motion vector information nor DCT information is available to the decoder. These macroblocks are called skipped macroblocks and are indicated when the macroblock_address_increment is greater than 1."). "In P-pictures, the skipped macroblock is defined to be a macroblock with a reconstructed motion vector equal to zero and no DCT coefficients" and "[i]n B-pictures, the skipped macroblock is defined to have the same macroblock_type (forward, backward, or both motion vectors) as the prior macroblock, differential motion vectors equal to zero, and no DCT coefficients." MPEG-1, §2.4.4.4. The figure below illustrates the relationship between an MPEG-1 skip macroblock (MV) and the prior macroblock used for predicting the skipped macroblock's motion vector (MV1). Saber, ¶184.



Coding for B-Pictures takes advantage of the observation in MPEG-1 that "[t]he motion vector of a macroblock tends to be well correlated with the vector of

previous macroblock. For example, in a pan all vectors would be roughly the same." MPEG-1, §D.6.2.3.

MPEG-1 also teaches that these macroblocks are integrated in typical video encoding and decoding techniques, but "[f]irstly, for a given macroblock, the coding mode is chosen." MPEG-1, §0.3. The coding mode "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." MPEG-1, §0.3. This prediction is used to form an error signal for a current block on which a transform and quantization is performed and then the quantized coefficient data are encoded. MPEG-1, §§0.3, D.2.2.3.

2. H.263

H.263 is titled "Video coding for low bit rate communication." The revision of H.263 referenced herein (EX-1009) was ratified by the ITU-T in February 1998, and made available to the public no later than May 11, 2000. H.263, i; Munford, 7-8.

H.263 is a video compression standard that "specifies a coded representation that can be used for compressing the moving picture component" of video. H.263, i. H.263 describes "a hierarchical structure with four primary layers," that include a

(1) picture; (2) Group of Blocks (GOB), or slice, or video picture segment; (3) macroblock; and (4) block. H.263, 13.

H.263 describes INTER coding (also called a P-picture), where a picture and/or macroblocks therein are predicted using a temporally previous reference picture, and INTRA coding (also called an I-picture), where no temporal prediction is applied. H.263, 11. When INTER coding is augmented by motion compensation, the decoder will by default accept one motion vector per macroblock.[14] H.263, 12.

When using motion compensation, H.263 specifies how a macroblock's motion vector "is obtained by adding predictors to the vector differences indicated by [Motion Vector Data or MVD]." H.263, 44. MVD "represents a pair of difference values" (e.g., a horizontal component difference and a vertical component difference), and is encoded in the macroblock data received by a decoder (e.g., from an encoder). H.263, 33-34, 38-40, 45, Fig. 10. The other component used to obtain a macroblock's motion vector, the predictor, "is the median value of the three

[14] H.263 also uses the terms "vector" and "macroblock vector" when discussing motion compensation. In H.263 sections pertaining to motion compensation, a POSITA would have understood *vector* and *macroblock vector* generally mean *motion vector* unless otherwise specified. Saber, ¶¶187-188.

candidate predictors," where the three candidate predictors "are taken from three surrounding macroblocks" neighboring the macroblock being decoded. H.263, 44-45. This scenario is annotated blue in H.263's Figure 12 below.

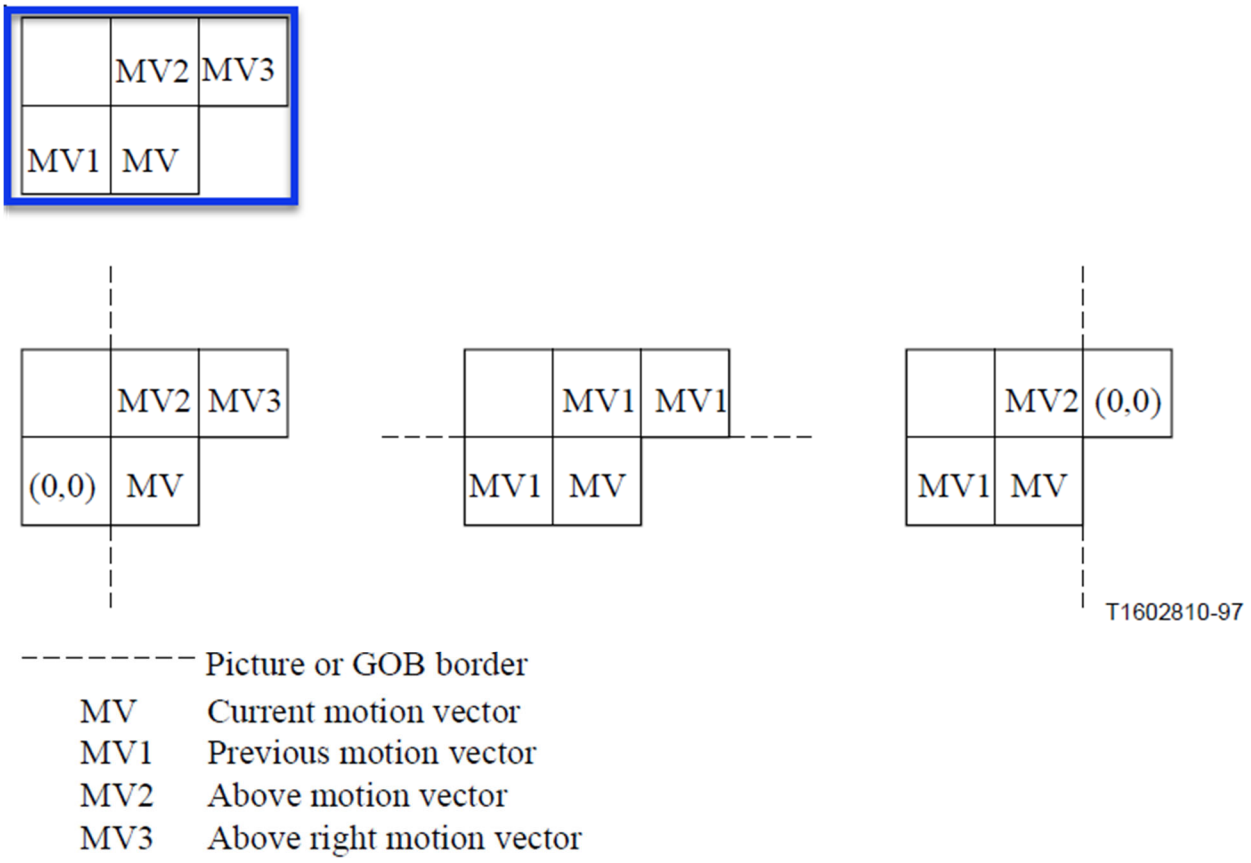


Figure 12/H.263 – Motion vector prediction

Thus, H.263 describes forming a predictor for a macroblock based on the motion vector (candidate predictors) of neighboring macroblocks. H.263 therefore teaches using motion vectors of neighboring macroblocks to determine the motion vector of a macroblock being decoded. Saber, ¶189. And as described in greater detail below, H.263 provides various rules for specifying the candidate predictors

corresponding to neighboring macroblocks to account for special cases, such as when a neighboring macroblock was INTRA coded and/or fell outside the boundaries of the current slice or picture. H.263, 44-45.

Although H.263 describes forming a macroblock's motion vector using predictors, and the rules for calculating the predictors, in the context of H.263's decoding process (H.263, 44), it would have been obvious to a POSITA that an H.263 encoder calculates predictors based on candidate predictors, to form a macroblock's motion vector, using the same rules. Saber, ¶190. H.263 describes how a decoder obtains a macroblock vector "by adding predictors to the vector differences indicated by MVD," where the MVD is part of the macroblock data (e.g., received from an encoder). H.263, 33-34, 38-40, 44, Fig. 10. Because the encoder provides the MVD to be used by the decoder to obtain a macroblock vector, both the encoder and decoder must perform the same steps for calculating the predictors to be used in combination with the MVD. Saber, ¶190. Otherwise, the decoder would obtain a different macroblock vector than the one obtained and used by the encoder for encoding the macroblock, despite the MVD being the same at both the encoder and decoder. *Id.* 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 calculating predictors

from candidate predictors, and the handling of the special cases therein, apply equally to H.263 encoders. Saber, ¶190.

3. Motivation to combine MPEG-1 and H.263

A POSITA would have been motivated to modify MPEG-1's coding of skipped macroblocks in B-Pictures to utilize a median prediction technique based on three surrounding previously-coded macroblocks, as taught by H.263, in order to improve the accuracy of the motion prediction. Saber, ¶¶191-195.

Use of a median in image coding was well-known by the time of the alleged invention in the '808 Patent, and provided benefits to image quality over techniques utilizing a single macroblock. For example, even before MPEG-1 was published in 1993, median filtering based on 3x3 blocks of pixels was used to improve image quality as compared to low-pass filtering techniques based on a single pixel. Saber, ¶191. However, a trade-off for the improved image quality provided by use of median is that it was more computationally intensive (and therefore also took longer) than techniques based on a single image segment. *Id*. For example, median filtering techniques required first sorting pixels before filtering. *Id.*

By 1998, H.263 utilized a median prediction technique to improve motion compensated prediction, as outlined in §VII.B.2. Advancements in computing in the approximately 5 years between MPEG-1 and H.263 made use of median techniques more attractive based on a number of factors including increased

processing capacity and speed, as well as lower cost for processing hardware. Saber, ¶192. This trend of increased processing power at a decreased cost continued to the date of the alleged invention in 2002, and so by then it would have been even more attractive to a POSITA to use a median prediction technique such as taught by H.263 in motion compensation because the cost of additional computational complexity was offset by advancements in computing hardware. *Id.*

Using H.263's median prediction technique for predicting skipped macroblocks in MPEG-1's B-pictures would have improved the accuracy of MPEG-1's skipped macroblock prediction because it uses three data points as compared to MPEG-1's single data point. Saber, ¶193. Use of three data points would have resulted in a better picture by reducing noise and preserving discontinuities in the motion compensated picture. *Id.*

Using H.263's median prediction technique would have also improved coding efficiency by making skipped macroblocks available in more scenarios. Saber, ¶194. For example, MPEG-1 does not allow for a skipped macroblock to follow an intra-coded macroblock. MPEG-1, §2.4.4.4. However, utilizing H.263's median prediction technique would allow for a skipped macroblock immediately following an intra-coded macroblock, as taught by H.263's first special case rule. H.263, 44-45.; Saber, ¶194.

Accordingly, a POSITA would have been motivated to improve MPEG-1's skipped macroblocks in B-Pictures by utilizing H.263's median prediction teachings, as opposed to simply copying the motion vector of preceding macroblock as taught by MPEG-1. Doing so would have amounted to no more than a simple substitution of one known element (MPEG-1's prediction based on a single previously coded macroblock) for another (H.263's median prediction based on three surrounding previously coded macroblocks) to obtain predictable results (improved, more accurate, video coding). Saber, ¶194. It also would have been use of a known technique (H.263's median prediction) to improve a similar method (MPEG-1's prediction based on a single macroblock) in the same way (predicting motion of a macroblock based on the motion of previously coded macroblocks). *Id*. It also would have been an application of a known technique (H.263's median prediction) to a known method (MPEG-1's single-macroblock prediction in skipped macroblocks in B-Pictures) ready for improvement (based on improvements in processing hardware) to yield predictable results (improved, more accurate, video coding). *Id.*

To the extent that PO argues that a POSITA would not have been motivated to use H.263's median prediction to improve the coding of MPEG-1's skipped macroblocks in B-Pictures because subsequent coding standards did not in fact do so, such an argument would be incorrect. Saber, ¶195. In fact, use of a median for

computing motion vectors was adopted in later standards such as H.264 under certain conditions. *Id.*; EX-1013, 176. Further, a POSITA would have understood that standards are the result of a widely collaborative effort among many stakeholders that sometimes have competing interests and ideas. Saber, ¶195. A POSITA would have understood that just because a possible solution was not entirely adopted in a standard, one would not necessarily be dissuaded from pursuing said solution if it provided a benefit. *Id.*

Accordingly, a POSITA would have been motivated to improve MPEG-1's coding of skipped macroblocks in B-Pictures by using H.263's median prediction technique to obtain the motion vector for a skipped macroblock rather than simply copying the motion vector from the immediately preceding macroblock, as taught by MPEG-1. Saber, ¶¶191-195.

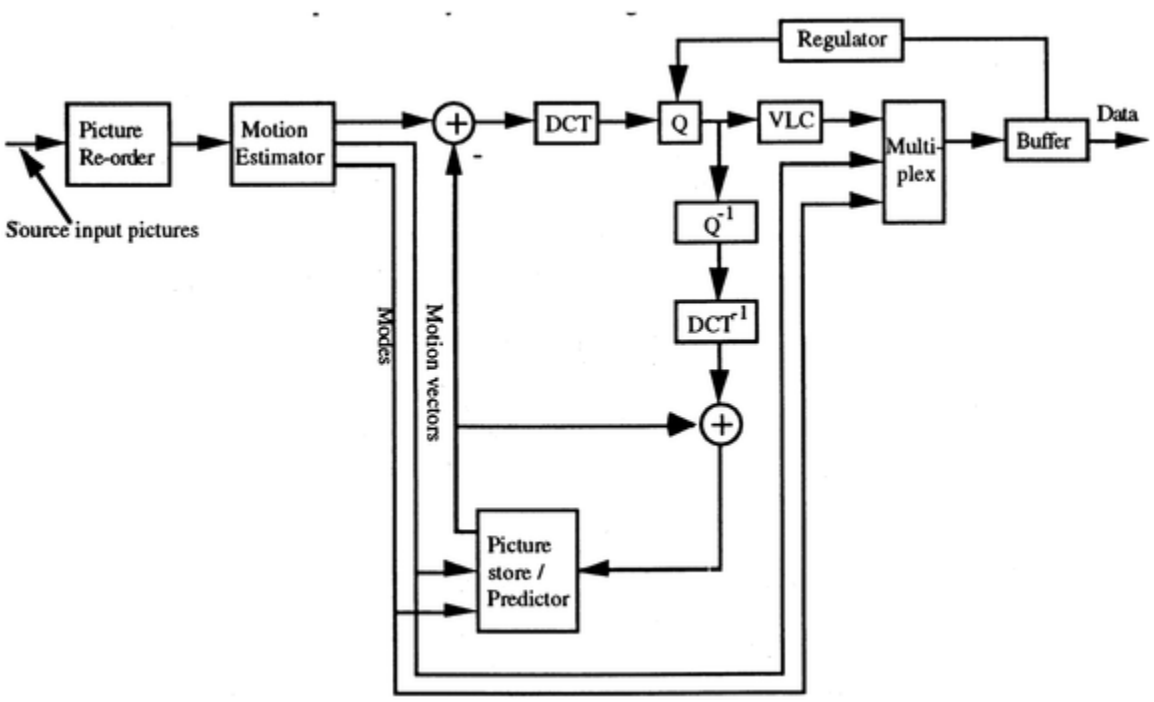
4. Claim 1 and Dependents

a. Claim 1

(i) 1[pre]

If limiting, MPEG-1 and H.263 render 1[pre] obvious.

MPEG-1 discloses a method of encoding a video sequence. For example, Figure 2 (below) details the various functions and blocks that constitute the video encoding process. "This clause is a brief description of the functions that need to be performed by an encoder." MPEG-1, Section 0.3 Encoding; *see also id.*, Annex D.



MPEG-1 at Figure 2 — Simplified video encoder block diagram

MPEG-1 explicitly outlines the digitization of the input video signal, preprocessing, format conversion, motion vector estimation, picture reordering, and coding of macroblocks sequentially within a picture. *Id.*, 0.3. Specifically, MPEG-1 states that the "encoder must choose which picture type to use for each picture," and "[h]aving defined the picture types, the encoder estimates motion vectors for each 16 by 16 macroblock in the picture," which constitutes a video encoding method. *Id.* Additionally, MPEG-1 describes the steps of choosing a coding mode, motion-compensated prediction, forming an error signal, DCT, quantization, and variable length coding, concluding with the regeneration of I-Pictures and P-Pictures

for further encoding. *Id.* All these elements described are part of a method of encoding a video sequence.

(ii) 1[a]

MPEG-1 and H.263 render 1[a] obvious. Saber, ¶¶198-206.

MPEG-1 defines "skipped macroblock" as a macroblock (first segment of the sequence) "for which no data are stored." MPEG-1, Section 2.1.131. MPEG-1 further clarifies that "[f]or some macroblocks there are no coded data, that is neither motion vector information nor DCT information is available to the decoder. These macroblocks are called skipped macroblocks and are indicated when the macroblock_address_increment is greater than 1." MPEG-1, Section 2.4.4.4. For P-pictures, "the skipped macroblock is defined to be a macroblock with a reconstructed motion vector equal to zero and no DCT coefficients." *Id.* For B-pictures, "the skipped macroblock is defined to have the same macroblock_type (forward, backward, or both motion vectors) as the prior macroblock, differential motion vectors equal to zero, and no DCT coefficients." *Id.* In other words, MPEG-1 teaches that for B-pictures, a macroblock assigned the skip mode is assigned the same motion vector as the immediately preceding macroblock.

As explained in §VII.B.2, H.263 teaches that a motion vector predictor for a current macroblock (MV) "is the median value for the three candidate predictors [MV1, MV2, and MV3]." H.263, 44-45, Fig. 12. H.263's Figure 12 shows the

standard case for motion vector prediction (blue) as well as special cases, some of which are relevant for dependent claims below:

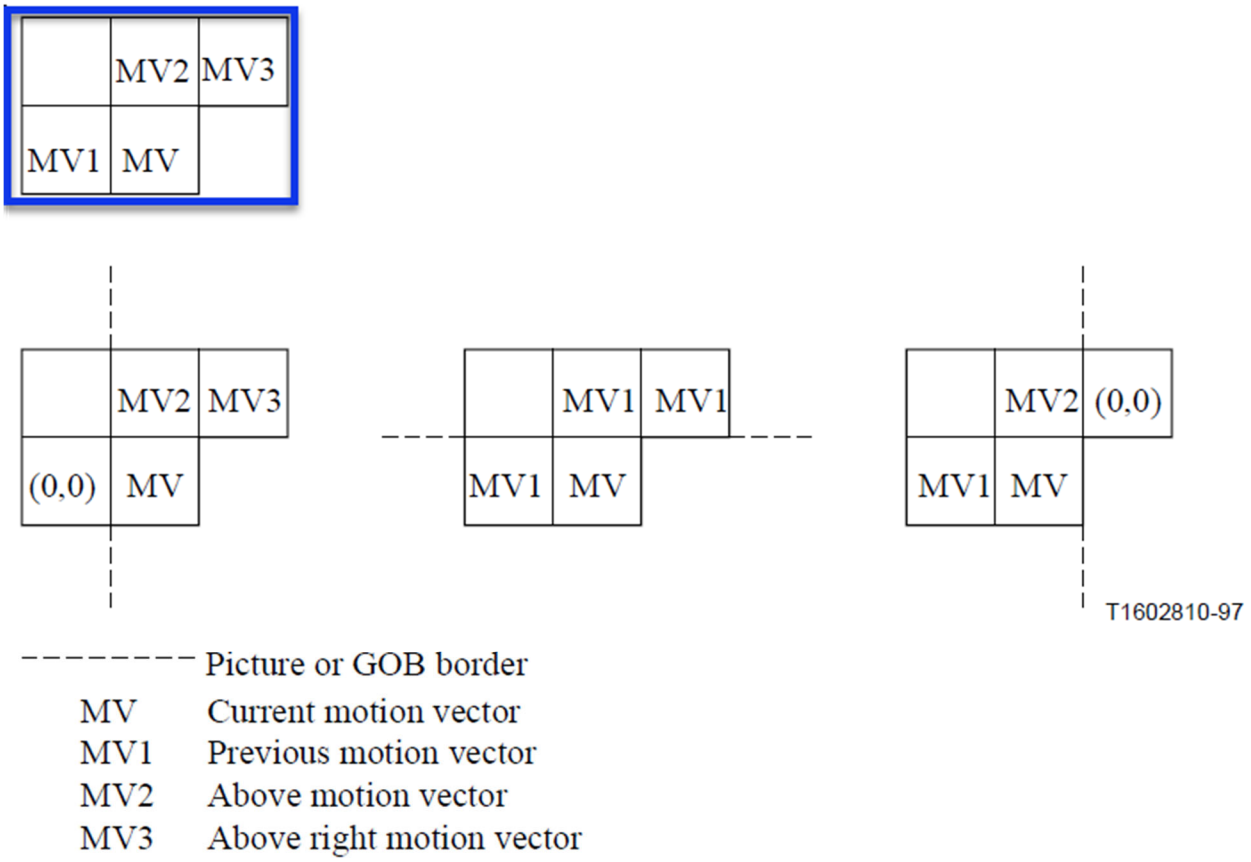


Figure 12/H.263 – Motion vector prediction

As further explained in §§VII.B.2 and VII.B.3, by the time of the alleged invention in the '808 Patent, a POSITA would have been motivated to improve MPEG-1's Skip mode by using H.263's median predictor teachings. Saber, ¶¶191-195. Using H.263's median predictor, rather than simply copying the motion vector of the prior macroblock, MPEG-1's Skip mode for B-pictures would have resulted in either a "zero (non-active) motion vector," e.g., in the case where the median of

MV1, MV2, and MV3 is 0, "or a non-zero (active) motion vector," e.g., in the case where the median of MV1, MV2, and MV3 is not zero, "[being] associated with each skip mode macroblock, depending on the motion in image segments surrounding the macroblock in question," i.e., MV1, MV2, and MV3. Saber, ¶¶201-202.

To the extent that PO argues that the above mapping reads does not satisfy the construction for *skip coding mode* because it does not teach analyzing the motion of surrounding macroblocks and assigning a zero (non-active) motion vector when the motion is below a given threshold, automatically generating a 0 motion vector for the macroblock, such an argument would be incorrect. *See*, IPR2024-00847, Paper 21, 54-55. First, neither the claim nor the construction of *skip coding mode* recites "automatically generating" a zero motion vector, or any language to that effect. Saber, ¶202. Second, the claim recites *assigning a skip coding mode*, and the construction for *skip coding mode* just requires "associate[ing]" a "zero (non-active) or non-zero (active) motion vector… depending on the motion in image segments surrounding the macroblock in question." The combination of MPEG-1 and H.263 renders obvious this limitation as explained above. PO's argument appears to try to import limitations from the specification based on disclosures related to, e.g., motion analysis performed by motion analysis block 802 in the '808 Patent's Figure 8. However, neither the Challenged Claims nor the claim construction of *skip coding*

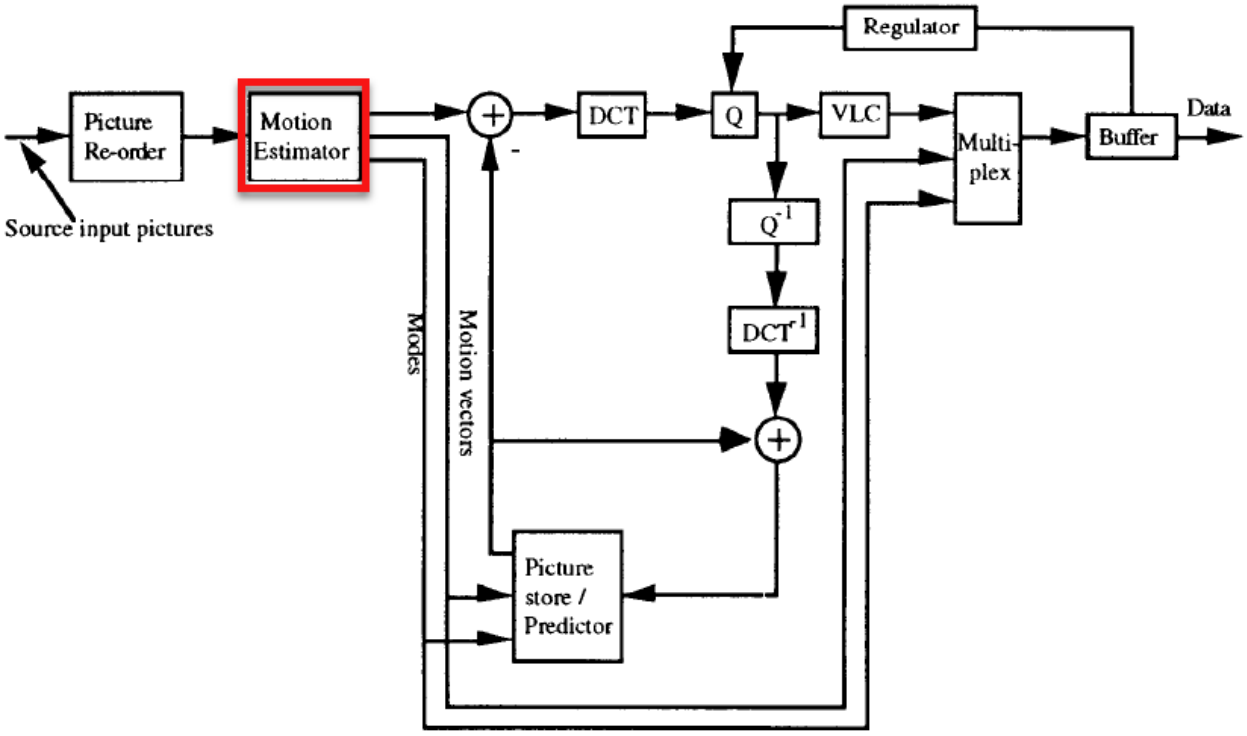
mode recite any form of "analysis." In fact, unchallenged claims 5, 8, 14, 19, 26, 37, 45, and 56 recite performing "analysis," so a POSITA would have understood that the Challenged Claims, which do not have that same recitation, are not so narrowed. Saber, ¶205. Such an understanding is also consistent with the '808 Patent's own teachings, which provide an embodiment in which surrounding motion analysis does not take place. '808pat, 24:42-45; Saber, ¶205.

In other words, MPEG-1 and H.263 render obvious *assigning a skip coding mode* (assigning a macroblock-address-increment greater than 1) *to a first segment of a first frame of the sequence* (a macroblock in a B-picture in MPEG-1), wherein the *skip coding mode* uses H.263's median predictor to determine the motion vector of the macroblock, such that *a zero (non-active) motion vector or a non-zero (active) motion vector is associated with each skip mode macroblock, depending on the motion in image segments surrounding the macroblock in question* (MV1, MV2, and MV3). Saber, ¶206.

(iii) 1[b]

MPEG-1 and H.263 render 1[b] obvious. Saber, ¶¶207-09.

Figure 2 of MPEG-1 in §0.3 outlines the main functional blocks of a simplified video encoder. MPEG-1, Section 0.3, Figure 2 (below).



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

The figure illustrates that the encoder includes a "Motion Estimator" block (red) which takes input from a "Picture Re-order" block and provides motion vectors. MPEG-1, Section 0.3. A POSITA would have understood that the Motion Estimator would assign motion vectors to segments of a frame, which could be either a zero motion vector (when no motion is estimated or for skipped segments) or a non-zero motion vector (when motion is predicted). Indeed, MPEG-1 describes that "in P-pictures one vector is needed for each non-intra macroblock and in B-Pictures

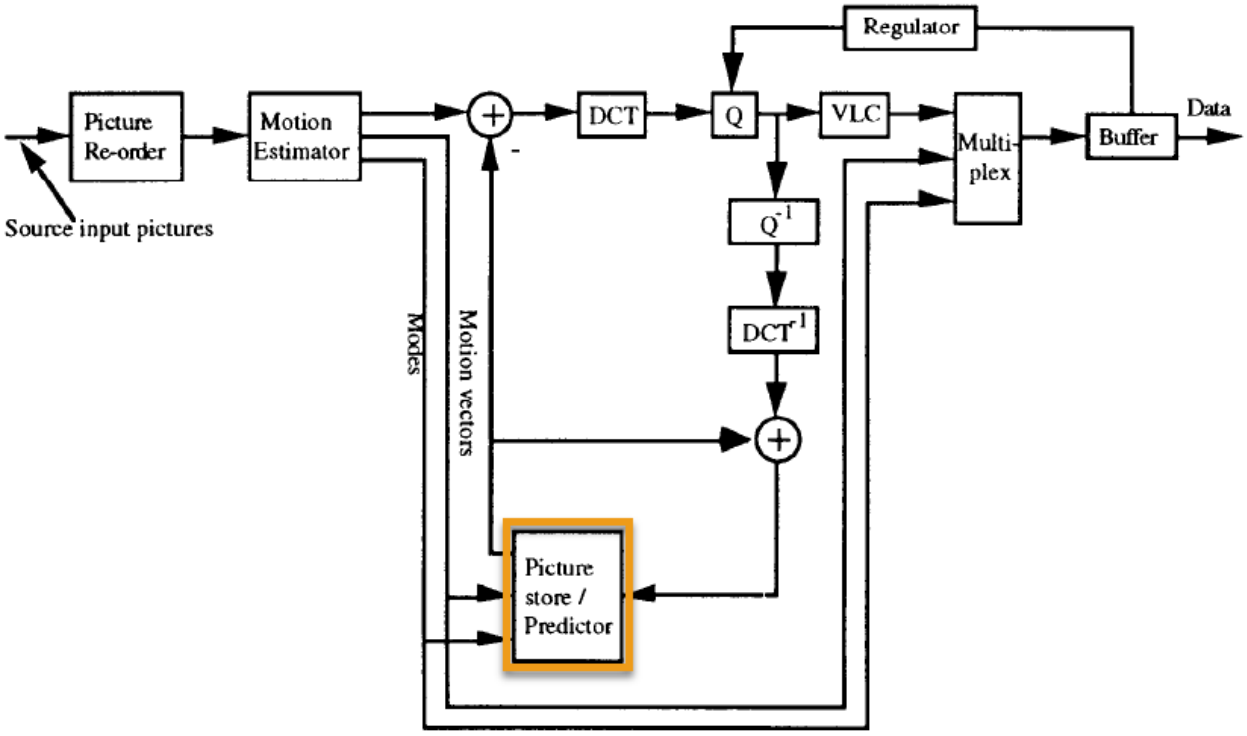
one or two vectors are needed," which indicates that the encoder assigns motion vectors to these macroblocks. Moreover, for B-pictures, MPEG-1 describes skipped macroblocks as having "the same macroblock-type as the prior macroblock, differential motion vectors equal to zero, and no DCT coefficients." In the MPEG-1/H.263 combination, rather than assigning the same motion vector as the prior macroblock, the motion vector is determined as the median of three surrounding macroblocks, as taught by H.263. In other words, if the median of the three surrounding macroblocks is zero, a zero motion vector is assigned, and if the median of the three surrounding macroblocks is not zero, a *predicted non-zero motion vector* is assigned to the skipped macroblock. Saber, ¶209. In either case, because the median of three surrounding macroblock motion vectors is used to assign the motion vector for the skip coding mode, it is *based at least in part on the motion information of a second segment neighboring the first segment.* Saber, ¶209.

(iv) 1[c]

MPEG-1 and H.263 render 1[c] obvious. Saber, ¶¶210-11.

As explained above §VII.B.4.a(iii), MPEG-1 discloses or renders obvious *assigning either a zero motion vector or a predicted non-zero motion vector for the skip coding mode for the first segment*. Afterwards, "a motion compensated prediction of the contents of the block based on past and/or future reference pictures is formed." MPEG-1, §0.3. Specifically, the assigned mode (here, the skip mode)

and motion vectors (*one of the zero motion vector and the predicted non-zero motion vector*) are then used by the "Picture store/Predictor" block (orange) to *predict* the content of the current frame segment by referencing the corresponding area in a previous or future frame. *Id.*



where

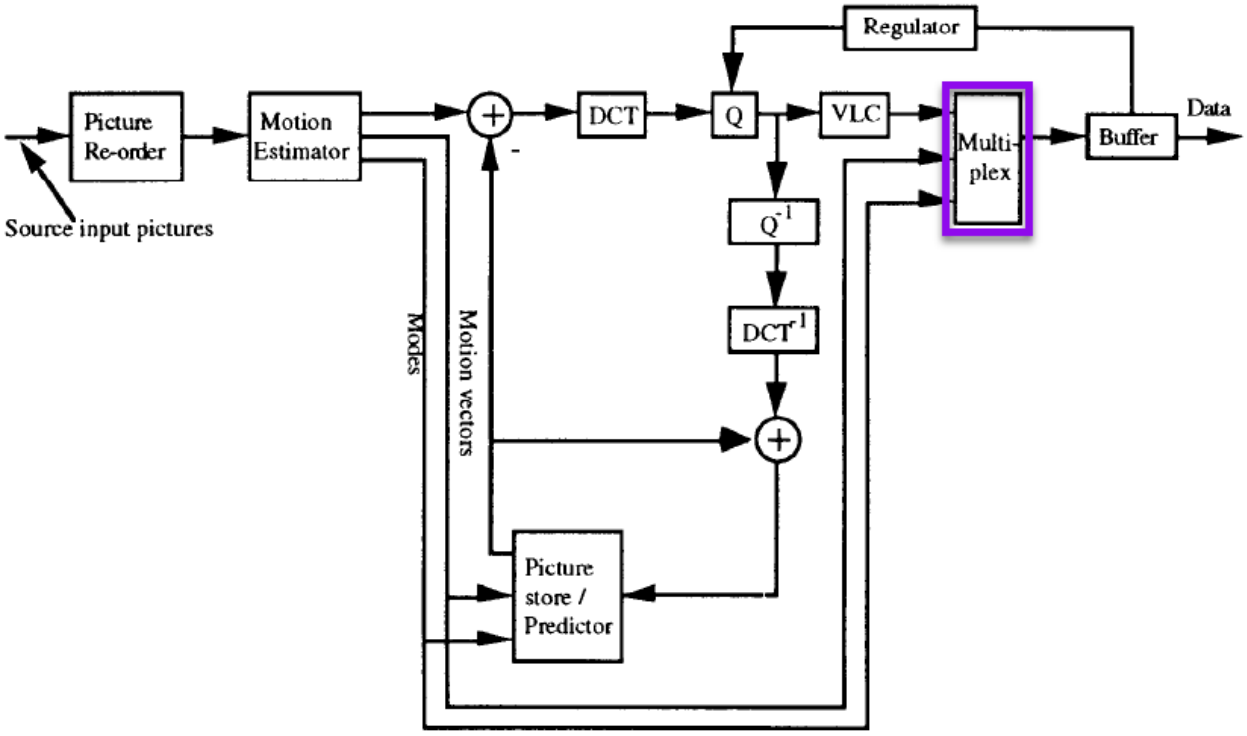
DCT is discrete cosine transform
DCT^{-1} is inverse discrete cosine transform
Q is quantization
Q^{-1} is dequantization
VLC is variable length coding

Figure 2 -- Simplified video encoder block diagram

(v) 1[d]

MPEG-1 and H.263 render 1[d] obvious. Saber, ¶¶212-13.

After a prediction is formed, as explained in § VII.B.4.a(iv), the prediction is used to create an error signal by "subtract[ing] [the prediction] from the actual data in the current macroblock." MPEG-1, §0.3. The error signal is transformed via a discrete cosine transform ("DCT") and quantized. *Id.* Then the quantized DCT coefficients and "the side information for the macroblock (mode, motion vectors, etc.)" are encoded using run-length coding. *Id.* Specifically, skipped macroblocks "are indicated when the macroblock_address_increment is greater than 1." MPEG-1, §2.4.4.4. As seen in Figure 2, the MPEG-1 encoder includes a Multiplex (purple) for combining various streams of data, including mode information, into an encoded bitstream. In other words, MPEG-1's multiplexer *provides in an encoded bitstream an indication of the skip coding mode* (the macroblock_address_increment being greater than 1) as part of the side information for the macroblock. *Id.*



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 further teaches that for skipped macroblocks "there are no coded data, that is neither motion vector information nor DCT information is available to the decoder." *Id.*, §2.4.4.4. Accordingly, *no further motion vector information for the first segment is coded in the encoded bitstream*. Saber, ¶213.

Thus, in the case of skipped macroblocks, the mode information indicating a skip would be part of what is encoded into the bitstream, without accompanying

motion vector data, since the motion vector for skipped macroblocks is zero or not changed from a prior macroblock. Saber, ¶¶212-213; MPEG-1, Section 2.4.4.4; *see also id.*, Section 0.3, Figure 2.

b. Claim 2

MPEG-1 and H.263 render obvious claim 2's additional limitation. Saber, ¶107.

MPEG-1 describes the characteristics of skipped macroblocks in B-pictures as "hav[ing] the same macroblock-type (forward, backward, or both motion vectors) as the prior macroblock, differential motion vectors equal to zero, and no DCT coefficients." MPEG-1, Section 2.4.4.4. Because the differential motion vectors are set to zero, it indicates that the motion vector of the skipped macroblock is presumed to be the same as that of the previous macroblock. In other words, in MPEG-1, the encoding of a skipped macroblock (first segment) is based on the characteristics of a macroblock that has been previously encoded (second segment). Additionally, the nature of video encoding, which is sequential and processes macroblocks in a raster scan order (left to right, top to bottom), means that MPEG-1's encoder would naturally handle macroblocks in close proximity to one another in sequence. Therefore, a POSITA would have understood that the motion characteristics of a previously encoded macroblock could logically be used in the predictive encoding

of a neighboring macroblock, especially in the context of skipped macroblocks in B-pictures as described in MPEG-1. Saber, ¶214; s*ee also* Section for limitation 1[b].

c. Claim 6

MPEG-1 and H.263 render obvious claim 6's additional limitation. Saber, ¶¶215-217.

As described in §VII.B.4.a(iii), H.263 describes a motion vector predictor with two components (horizontal and vertical), where "[f]or each component, the predictor is the median value of the three candidate predictors for this component." H.263, 44-45. H.263 further describes how "the candidate predictors for the differential coding are taken from three surrounding macroblocks as indicated in Figure 12." H.263, 44. Thus, referring to H.263's Figure 12, a motion vector predictor for a current macroblock (MV in the figure) is calculated as the median of the motion vectors of the neighboring left, top, and top-right macroblocks (MV1, MV2, MV3 in the figure, respectively). Saber, ¶217. H.263's Figure 12 highlights (blue) the calculation described above, where a macroblock's motion vector (predictor) is calculated as the component-wise median of its three neighboring macroblocks' motion vectors (candidate predictors). Saber, ¶216.

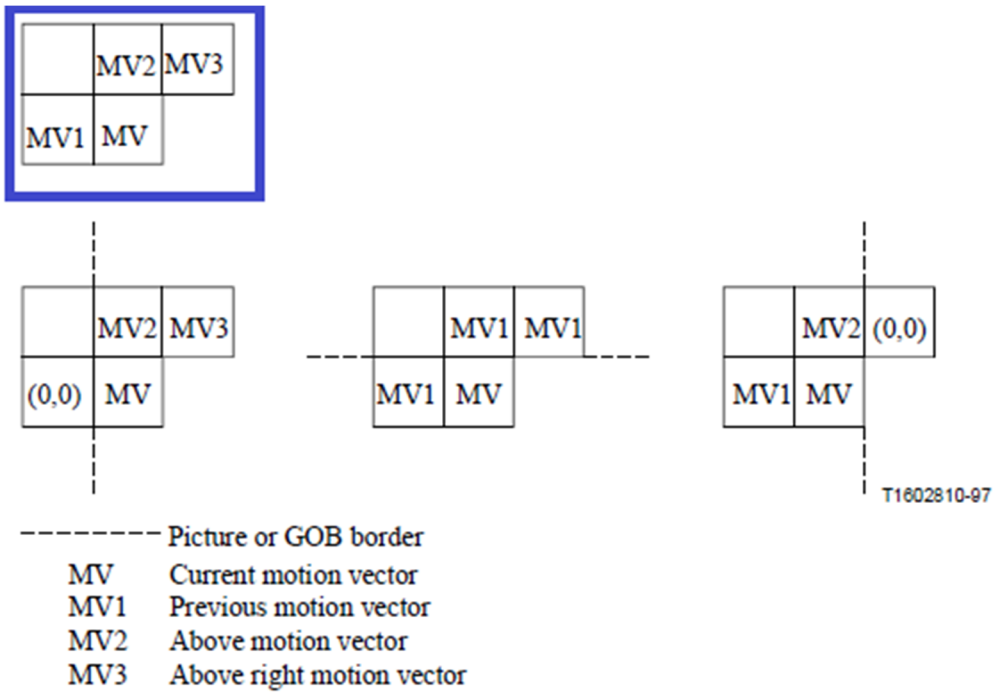


Figure 12/H.263 – Motion vector prediction

H.263, Fig. 12.

H.263 also specifies "decision rules" to be applied for neighboring candidate predictors in the "special cases at the borders of the current GOB, slice, or picture." H.263, 44. The third of the decision rules specifies "the candidate predictors MV2 and MV3 are set to MV1 if the corresponding macroblocks are outside the picture (at the top) or outside the GOB (at the top) if the GOB header of the current GOB is non-empty." H.263, 45. That is, in H.263's median predictor, when a current macroblock is at the top border of a picture or GOB, by application of the third decision rule, the candidate predictors provided by the motion vectors of the top and top-right neighboring macroblocks (MV2 and MV3, respectively) instead are set to

be the motion vector of the left neighboring macroblock (MV1). Saber, ¶¶217. Thus, all three candidate predictors are set as the left neighboring macroblock's motion vector. *Id*. H.263's Figure 12 highlights (green) the application of the third decision rule. *Id.*

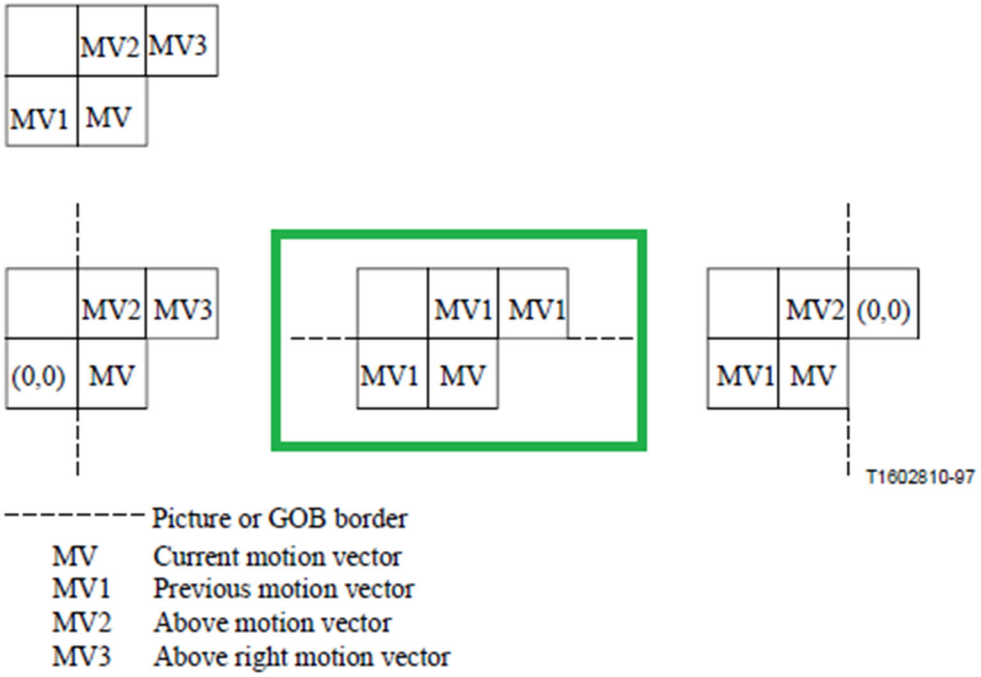


Figure 12/H.263 – Motion vector prediction

H.263, Fig. 12.

A POSITA would have known the median of three numbers, all of which have the same value, is equal to that value. *See* Saber, ¶217. Thus, when the three candidate predictors are the motion vector of the left neighboring macroblock (MV1) due to application of the third decision rule, the predicted motion vector of the

current macroblock (MV), which is calculated as the median of the candidates, equals MV1 as well. *See* Saber, ¶217.

H.263 thus teaches a motion vector predicted for a macroblock may equal the motion vector of a neighboring macroblock (e.g., of the left neighbor, when the macroblock is at a top boundary). When that occurs, if the neighboring macroblock's motion vector (MV1) is a zero-motion vector, then the current macroblock's predicted motion vector (MV) will also be the zero-motion vector. Saber, ¶217. H.263 therefore teaches or suggests "*if the second segment has [ZMV], the [ZMV] is assigned.*"

d. Claim 23

(i) 23[pre]

If limiting, MPEG-1 and H.263 render 23[pre] obvious. Because H.263 teaches predicting a motion vector as the median value of three candidate predictors, a POSITA would have understood that, at least in some instances, a second segment surrounding the macroblock in question has a non-zero motion vector. Saber, ¶218.

(ii) 23[a]

MPEG-1 and H.263 render 23[a] obvious. Saber, ¶91. As explained in §VII.B.4.a(ii), MPEG-1 using H.263's median predictor derives a motion vector by taking the median of three macroblocks. H.263, 44-45. In certain cases, a predicted non-zero motion vector would be derived through this operation. For example, if all

three macroblocks have non-zero motion vectors, a *predicted non-zero motion vector* would be derived by taking the median of the motion vectors of the three surrounding macroblocks, including the *second segment*. Alternatively, in the scenario discussed in §VII.B.4.c, if MV1 is (*second segment*) is not zero, under the third rule disclosed by H.263, MV2 and MV3 are set to the value of MV1, resulting in a *predicted non-zero motion vector based at least in part on the motion vector of the second segment* (MV1).

(iii) 23[b]

MPEG-1 and H.263 render 23[b] obvious. Saber, ¶221. As explained in §VII.B.4.a(iii), MPEG-1 using H.263's median prediction assigns a predicted non-zero motion vector to a skipped macroblock in B-pictures where the median motion vector of the three surrounding macroblocks is non-zero. Saber, ¶221.

(iv) 23[c]

MPEG-1 and H.263 render 23[c] obvious. Saber, ¶222. As explained in §VII.B.4.d(iv), MPEG-1 teaches that after a mode and/or motion vector is assigned to a macroblock, "a motion compensated prediction of the contents of the block based on past and/or future reference pictures is formed" using said assigned information for the macroblock. MPEG-1, §0.3. In other words, MPEG-1's predictor forms a prediction of the skipped macroblock (*the prediction for the first segment*) with respect to a past and/or future frame (*with respect to the reference frame*) based

on the assigned mode information, in this case a skipped macroblock assigned a non-zero motion vector (*based at least in part on the predicted non-zero motion vector*).

e. Claim 24

MPEG-1 and H.263 render obvious claim 24's additional limitation. Saber, ¶¶223-24.

As described in §VII.B.4.c, the MPEG-1-H.263 combination teaches *if the second segment has [ZMV], the [ZMV] is assigned for skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the [ZMV].* The MPEG-1-H.263 combination teaches the substantially similar limitation of this claim. Saber, ¶223.

The MPEG-1-H.263 combination teaches the *second segment is predicted using motion-compensated prediction from the reference picture*, as additionally recited. Saber, ¶224. As described in §VII.B.4.b, MPEG-1 teaches that the second segment is a "previously encoded segment." A POSITA would have understood that, as long as the recited *second segment* is not located in the upper left corner of an image (such that it would have no prediction candidates), in some cases the *second segment is predicted using motion-compensated prediction from the reference picture* and could have been coded with a zero motion vector. E.g., §§VII.B.4.a(iv), VII.B.4.d(iv); Saber, ¶224.

f. Claim 25

MPEG-1 and H.263 render obvious claim 25's additional limitation for the same reasons discussed in §VII.B.4.e, and because the MPEG-1-H.263 combination additionally teaches the second reference picture is *immediately preceding the picture second segment belongs to*, as additionally recited by claim 25. Saber, ¶225.

MPEG-1 teaches that a macroblock in a B-picture may be constructed "from a previous picture," which a POSITA would have understood to include a reference picture *immediately preceding the picture second segment belongs to*. MPEG-1, §D.6.5.2.1; Saber, ¶225. H.263 also describes using "the most recent temporally previous anchor picture," which a POSITA would have understood is typically the immediately preceding frame in H.263 (e.g., when optional B-pictures are not used). Saber, ¶225; H.263, 11-12, 30. Therefore the MPEG-1-H.263 combination teaches the second reference picture may immediately precede the picture to which the claimed second segment belongs. Saber, ¶225.

g. Claim 28

MPEG-1 and H.263 render obvious claim 28's additional limitation. Saber, ¶226. As explained in §VII.B.4.a(ii), H.263's median predictor derives a motion vector as the median of previously coded surrounding macroblocks. The predicted non-zero motion vector is therefore derived *based at least in part on the motion vector of the second segment.* Saber, ¶226.

h. Claim 29

MPEG-1 and H.263 render obvious claim 29's additional limitation. Saber, ¶227. As explained in §VII.B.4.a(ii), H.263's median predictor derives a motion vector as the median of the motion vectors of three previously coded surrounding macroblocks. The predicted non-zero motion vector is therefore derived *based at least in part on the motion vector of the second segment and motion vector of a third segment neighboring the first segment*. Saber, ¶227.

i. Claim 30

MPEG-1 and H.263 render obvious claim 30's additional limitation. Saber, ¶228. MPEG-1 explicitly defines a skipped macroblock as a "macroblock for which no data are stored," which indicates that for skipped macroblocks, the encoding process does not generate any residual data—such as differences between the predicted macroblock and the actual macroblock—that would be stored in the bitstream. MPEG-1, §2.1.131, §2.4.4.4 ("[f]or some macroblocks there are no coded data, that is neither motion vector information nor DCT information is available to the decoder."). Thus, in MPEG-1, when a macroblock is skipped, no residual data (which would have been represented by DCT coefficients after transformation and quantization of the prediction error) is provided in the bitstream. Saber, ¶228.

5. Claim 7 and Dependents

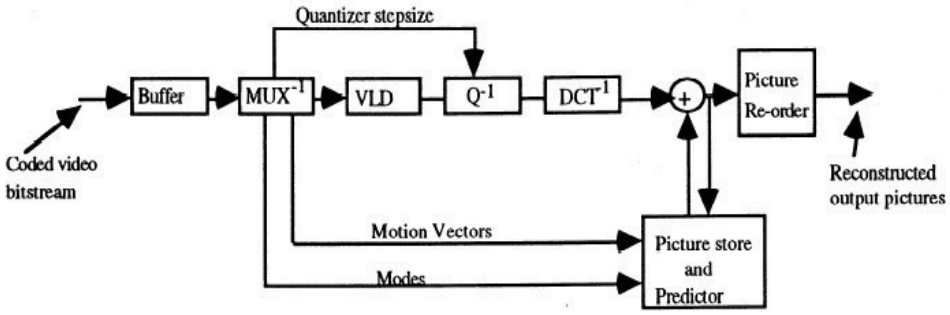
MPEG-1 discloses that after a coded representation of a picture is generated by the encoder, the encoder "decod[es] the data so that [the decoded pictures] can be used as reference pictures for the subsequent encoding." MPEG-1, §0.3. MPEG-1 further discloses that "[d]ecoding is the inverse of the encoding operation." *Id.*, §0.4. Accordingly, both an MPEG-1 encoder and decoder perform *a method of decoding*…which is the inverse of the encoding method discussed *infra*.

a. Claim 7

(i) 7[pre]

If limiting, MPEG-1 and H.263 render 7[pre] obvious. Saber, ¶¶230-32.

MPEG-1 explains that the decoding process is "the inverse of the encoding operation." MPEG-1, §04. "The decoder reads the bitstream, it identifies the start of a coded picture and then the type of the picture…[and] decodes each macroblock in the picture in turn." *Id.* This process of reading the bitstream and identifying picture starts and types includes the decoding of an encoded video sequence. MPEG-1's Figure 4 and related disclosures outline a decoder and decoding process. *Id.*



Where

DCT^{-1}	is inverse discrete cosine transform
Q^{-1}	is dequantization
MUX^{-1}	is demultiplexing
VLD	is variable length decoding

Figure 4 -- Basic video decoder block diagram

Similarly, Figure D.7 demonstrates the different components a decoder might use to process inter-coded frames (P- or B-pictures), showing how motion compensation (MC) is applied to the reference frames stored in the decoder to aid in the reconstruction of the current frame. *Id.*, D.2.7.

A simplified block diagram of a possible decoder implementation is shown below:

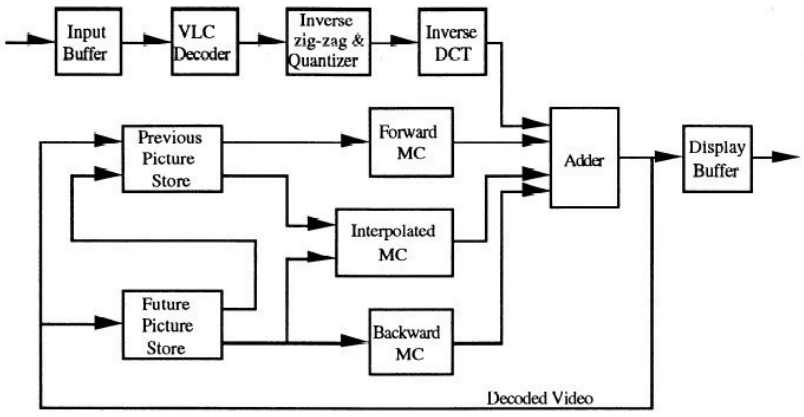


Figure D.7 -- Simplified decoder block diagram

Both Figures 4 and D.7 from MPEG-1 illustrate the blocks of a basic video decoder, showing components like the "Buffer," "VLD" (Variable Length Decoder), "Inverse Quantizer," and "Inverse DCT," among others. These diagrams offer a

visual representation of the decoder's architecture and steps used to decode an encoded video sequence. Saber, ¶¶230-32.

(ii) 7[a]

MPEG-1 and H.263 render 7[a] obvious. Saber, ¶233. As explained in §VII.B.4.a(v), the coding mode for a macroblock is encoded by MPEG-1's encoder in an encoded bitstream. When an MPEG-1 decoder receives the encoded bitstream, it includes the indication of the skip coding mode encoded therein, which in MPEG-1 is indicated "when the macroblock_address_increment is greater than 1." MPEG-1, §2.4.4.4; §0.4 ("the decoder reads the bitstream, it identifies the start of a coded picture and then the type of the picture. It decodes each macroblock in the picture in turn. The macroblock type and the motion vectors, if present, are used…").

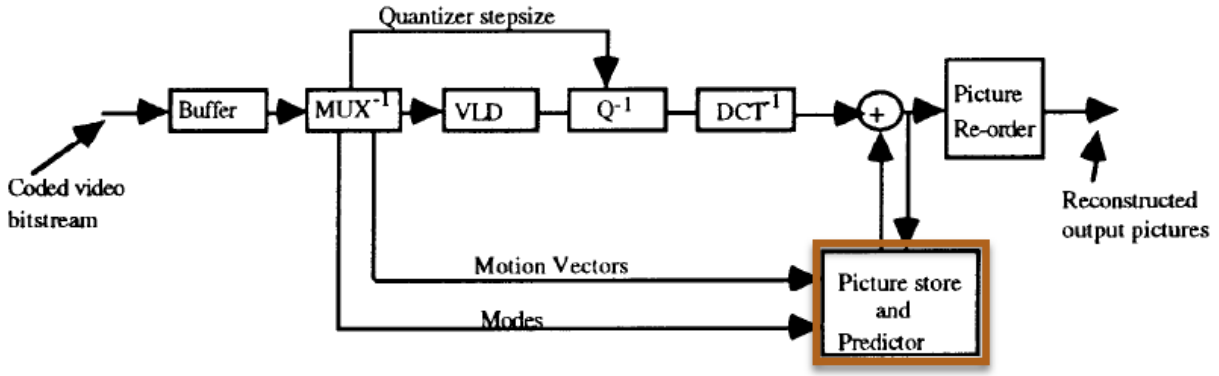
(iii) 7[b]

MPEG-1 and H.263 render 7[b] obvious for the reasons discussed in §VII.B.4.a(ii) because MPEG-1's decoding method performs the same steps as the encoding method for assigning either a zero motion vector or a predicted non-zero motion vector for the skip coding mode. Saber, ¶234.

(iv) 7[c]

MPEG-1 and H.263 render 7[c] obvious for the reasons discussed in §VII.B.4.a(iv) for forming a prediction. Saber, ¶235. MPEG-1 explicitly teaches that "the macroblock type and the motion vectors, if present, are used to construct a

prediction of the current macroblock based on past and future reference pictures that have been stored in the decoder." MPEG-1, §0.4, Fig. 4 (depicting a picture store and predictor (brown)).



Where

DCT^{-1}	is inverse discrete cosine transform
Q^{-1}	is dequantization
MUX^{-1}	is demultiplexing
VLD	is variable length decoding

Figure 4 -- Basic video decoder block diagram

b. Claim 9

MPEG-1 and H.263 render obvious claim 9's additional limitation. Saber, ¶236.

The MPEG-1-H.263 combination teaches *wherein if the second segment has [ZMV], the [ZMV] is assigned for skip coding mode of the first segment and the prediction for the first segment is formed with respect to a corresponding segment of the reference frame associated with the [ZMV]* (§VII.B.4.c), *a second segment neighboring the first segment* (§VII.B.4.a(iii)), *the second segment is a previously encoded segment neighboring the first segment* (§VII.B.4.b).

Thus, the second segment neighboring the first segment, as taught or suggested by the MPEG-1-H.263 combination, is in a previously encoded *region surrounding the first segment*. Saber, ¶236. Further, a POSITA would have understood a decoding method decodes blocks or segments in a bitstream in the same order they were encoded and placed in the bitstream, and therefore the MPEG-1-H.263 combination additionally teaches the claimed second segment is *in a previously decoded region surrounding the first segment*, as recited by claim 9. Saber, ¶236.

c. Claim 32

MPEG-1 and H.263 render obvious claim 32's additional limitation for the same reasons discussed in §§VII.B.4.c, VII.B.5.a(i), because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶237.

d. Claim 34

MPEG-1 and H.263 render obvious claim 34's additional limitation for the same reasons discussed in §§VII.B.4.d-VII.B.4.d(iv), because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶238.

e. Claim 36

MPEG-1 and H.263 render obvious claim 36's additional limitation for the same reasons discussed in §§VII.B.4.e, because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶239.

f. Claim 39

MPEG-1 and H.263 render obvious claim 39's additional limitation for the same reasons discussed in §VII.B.4.g, because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶240.

g. Claim 40

MPEG-1 and H.263 render obvious claim 40's additional limitation for the same reasons discussed in §VII.B.4.h, because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶241.

h. Claim 41

MPEG-1 and H.263 render obvious claim 41's additional limitation for the same reasons discussed in §§VII.B.4.i, because no residual information is provided for the first segment in the encoded bitstream from the encoder for MPEG-1 skipped macroblocks. Saber, ¶242.

i. Claim 43

MPEG-1 and H.263 render obvious claim 43's additional limitation for the same reason discussed in §VII.B.4.f, because the combined decoding method performs the same steps as the encoding method for forming a prediction of a segment. MPEG-1, §0.4; Saber, ¶243.

j. Claim 65

MPEG-1 and H.263 render obvious claim 65's additional limitation for the same reasons discussed in §VII.B.4.a(v), because MPEG-1 does not include any further motion vector information with skipped macroblocks. MPEG-1, §2.4.4.4; Saber, ¶244.

VIII. CONCLUSION

Inter Partes Review of the Challenged Claims is respectfully requested.[15]

[15] Currently, the memorandum on interim processes for PTAB workload management applies, and the Office has stated the petition should not address discretionary issues; instead, those issues should be raised in opposition to PO's discretionary denial brief (if filed). *See Interim Processes for PTAB* Coke Stewart, *Workload Mgmnt.*, (Mar. 26, 2025), https://www.uspto.gov/patents/ptab/faqs/interim-processes-workload-management.

IX. MANDATORY NOTICES

A. Real Party in Interest

Petitioners identify themselves as real parties in interest. Out of an abundance of caution, Petitioners also disclose Element Electronics as a potential real party in interest.

B. Related Matters

To the best of Petitioner's knowledge, the '808 Patent has been involved in the following matters:

- *Nokia Technologies Oy v. Amazon.com, Inc., et al.*, 1-23-cv-01236 (D. Del. Oct. 31, 2023)
- *Nokia Technologies Oy v. HP Inc.*, 1-23-cv-01237 (D. Del. Oct. 31, 2023)
- *Nokia Corporation et al. v. Amazon.com, Inc. et al.*, Case No. 1:23-cv-01232 (D. Del. Oct. 27, 2023)
- *Nokia Technologies Oy v. Lenovo (Shanghai) Electronics Technology Co. Ltd. et al.*, Case No. 5:19-cv-00427 (E.D.N.C. Sept. 25, 2019)
- *Nokia Technologies Oy v. Apple Inc.*, Case No. 2:16-cv-01440 (E.D. Tex. Dec. 21, 2016)
- *Certain Electronic Devices, Including Computers, Tablet Computers, and Components and Modules Thereof*, Inv. No. 337-TA-1208 (ITC July 2, 2020)

- *Certain Video Capable Electronic Devices, Including Computers, Streaming Devices, Televisions, and Components and Modules Thereof*, Inv. No. 337-TA-1380 (ITC Oct. 31, 2023)

C. Notice of Counsel and Service Information

LEAD COUNSEL
Jessica Kaiser (Reg. No. 58,937) kaiser-ptab@perkinscoie.com Perkins Coie LLP 1900 Sixteenth Street, Suite 1400 Denver, CO 80202-5255 Telephone: (303) 291-2300
BACK-UP COUNSEL
Jon R. Carter, Reg. No. 75,145 PERKINS COIE LLP 1155 Avenue of the Americas 22nd Floor New York, NY 10036-2711 Tel: (212) 262-6900 carter-ptab@perkinscoie.com Atanas Baitchev, Reg. No. 82,414 PERKINS COIE LLP 11452 El Camino Real Suite 300 San Diego, CA 92130-2080 Tel: (858) 720-5700 baitchev_ptab@perkinscoie.com

Respectfully submitted,

/ *Jessica Kaiser* /

Jessica Kaiser
Reg. No. 58,937
Attorney for Petitioner

PERKINS COIE LLP
1900 Sixteenth Street, Suite 1400
Denver, CO 80202-5255

Date: June 13, 2025

CERTIFICATE OF WORD COUNT UNDER 37 CFR §42.24(D)

Pursuant to 37 C.F.R. §42.24(a), Petitioner hereby certifies that portions of the above-captioned Petition for *Inter Partes* Review of U.S. Patent 7,532,808, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 12,951. Pursuant to 37 C.F.R. §42.24(a), this word count is in compliance and excludes the table of contents, table of authorities, mandatory notices under §42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing. This word count was prepared using Microsoft Word.

Respectfully submitted,

/ *Jessica Kaiser* /

Jessica Kaiser
Reg. No. 58,937
Attorney for Petitioner

Date: June 13, 2025
PERKINS COIE LLP
1900 Sixteenth Street, Suite 1400
Denver, CO 80202-5255

CERTIFICATE OF SERVICE

The undersigned hereby certifies that true copies of the Petition for *Inter Partes* Review of U.S. Patent No. 7,532,808 and supporting materials (Exhibits and Power of Attorney) were served via overnight delivery on the Patent Owner at the correspondence address of record as listed on PAIR:

WARE, FRESSOLA, MAGUIRE & BARBER LLP
55 Corporate Drive
3rd Floor
Trumbull, CT 06611

A courtesy copy was also sent via electronic mail to Patent Owner's litigation counsel listed below:

Peter Kohlhepp
pkohlhepp@carlsoncaspers.com

Samuel T Lockner
slockner@carlsoncaspers.com

Tara C Norgard
tnorgard@carlsoncaspers.com

Respectfully submitted,

/ *Jessica Kaiser* /

Jessica Kaiser
Reg. No. 58,937
Attorney for Petitioner

Date: June 13, 2025