

Petition for *Inter Partes* Review of USP 8,090,025

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re *Inter Partes* Review of:)
U.S. Patent No. 8,090,025)
Issued: January 3, 2012)
Application No.: 11/787,623)
Filing Date: April 17, 2007)

For: **Moving-Picture Coding Apparatus, Method and Program, and Moving-Picture Decoding Apparatus, Method and Program**

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 8,090,025

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	IDENTIFICATION OF CHALLENGES (37 C.F.R. § 42.104(b)).....	1
III.	BACKGROUND	2
A.	The '025 Patent (Ex. 1001)	2
B.	Prosecution History	6
C.	Person of Ordinary Skill in the Art	7
D.	Claim Construction	7
IV.	FOUNDATIONS 1-2.....	8
A.	Overview of the Prior Art.....	8
1.	Mualla (Ex.1005)	8
2.	Shirani (Ex.1006).....	26
3.	Combination of Mualla and Shirani.....	30
4.	Saito (Ex.1007)	32
5.	Combination of Mualla-Shirani and Saito	34
6.	Takenaka (Ex.1023).....	41
7.	Combination of Mualla-Shirani-Saito and Takenaka	43
B.	Claims 6-8, 10 (“Moving picture decoding”)	46
1.	[6.pre].....	47
2.	[6.1.DEC]	48
3.	[6.2.ENC]	51
4.	[6.3.ENC]	61
5.	[6.4.ENC]	63

6.	[6.5.ENC]	67
7.	[6.6.ENC]	68
8.	[6.7.ENC]	69
9.	[6.8.DEC]	71
10.	[6.9.DEC]	74
11.	[6.10.DEC]	76
12.	[6.11.DEC]	78
13.	[6.12.DEC]	81
14.	[6.13.DEC]	83
15.	[6.14.DEC]	85
16.	[6.15.DEC]	87
17.	[6.16.DEC]	91
18.	[7.pre]	95
19.	[7.1.DEC]-[8.3], [10.pre]-[10.13.DEC]	95
C.	Claims 1-5, 9 (“Moving picture coding”)	96
1.	[1.pre], [4.pre], [9.pre]	96
2.	[1.8.ENC], [2.3], [4.8.ENC], [5.3], [9.6.ENC]	100
3.	[3.pre]	100
4.	[3.1]	101
5.	[3.2]	102
6.	Remaining limitations of Claims 1, 4, and 9	103
V.	THE BOARD SHOULD REACH THE MERITS OF THIS PETITION	104

A.	<i>Advanced Bionics</i> Favors Institution.....	104
B.	<i>General Plastic</i> Factors Favor Institution.....	105
C.	<i>Fintiv</i> Factors Favor Institution.....	106
VI.	MANDATORY NOTICES UNDER 37 C.F.R. § 42.8(a)(1).....	107
A.	Real Party-In-Interest	107
B.	Related Matters.....	108
C.	Grounds for Standing	108
D.	Counsel and Service Information.....	108
E.	IPR Fee	109
VII.	CONCLUSION.....	110

TABLE OF AUTHORITIES

Page(s)

CASES

<i>Advanced Bionics LLC v. MED-EL Elektromedizinische Gerate GmbH,</i> IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020).....	104
<i>Advanced Coding Technologies LLC v. Apple Inc.,</i> No. 2:24-cv-00572-JRG (E.D. Tex.)	108
<i>Advanced Coding Technologies LLC v. Google LLC,</i> No. 2:24-cv-00353-JRG (E.D. Tex.)	108
<i>Becton, Dickinson and Co. v. B. Braun Melsungen AG,</i> IPR2017-01586, Paper 8 (PTAB Dec. 15 2017)	105
<i>Ford Motor Co. v. Neo Wireless LLC,</i> IPR2023-00763, Paper 28 (PTAB Mar. 22, 2024)	106
<i>General Plastic Industrial Co., Ltd. v. Canon Kabushiki Kaisha,</i> IPR2016-01357, Paper 19 (PTAB Sept. 6, 2017).....	105
<i>KSR Int’l Co. v. Teleflex Inc.,</i> 550 U.S. 398, 82 USPQ2d 1385 (2007)	32, 39
<i>Samsung Elecs. Co., Ltd. v. Advanced Coding Techs. LLC,</i> IPR2024-00374, Paper 10 (PTAB Jul. 22, 2024).....	7, 107
<i>Valve Corp. v. Electronic Scripting Products, Inc.,</i> IPR2019-00062, Paper 11 (PTAB Apr. 2, 2019)	105
<i>Vivendum Prod. Sols., Inc. v. Rotolight Ltd.,</i> IPR2023-01218, Paper 12 (PTAB Apr. 19, 2024)	106

EXHIBIT LIST

Exhibit	Description
1001	U.S. Patent No. 8,090,025 to Sakazume (“the ’025 patent”)
1002	Excerpts from the Prosecution History of the ’025 patent (“the Prosecution History”)
1003	Declaration and Curriculum Vitae of Dr. Reader
1004	[reserved]
1005	Al-Mualla, et al., “Video Coding for Mobile Communications: Efficiency, Complexity and Resilience,” Elsevier, 2002 (“Mualla”)
1006	S. Shirani, et al., “A Concealment Method for Video Communications in an Error-Prone Environment,” in IEEE Journal on Selected Areas in Communications, vol. 18, no. 6, pp. 1122-1128, June 2000 (“Shirani”)
1007	N. Saito, et al., “The Polyharmonic Local Sine Transform: A New Tool for Local Image Analysis and Synthesis without Edge Effect,” Applied and Computational Harmonic Analysis, vol. 20, pp. 41-73, January 2006 (“Saito”)
1008	B. Girod, et al., “Feedback-Based Error Control for Mobile Video Transmission,” in Proceedings of the IEEE, vol. 87, no. 10, pp. 1707-1723, Oct. 1999 (“Girod”)
1009	T. Stockhammer, et al., “H.264/AVC in Wireless Environments,” in IEEE Trans. on Circuits and Systems for Video Tech., vol. 13, no. 7, July 2003 (“Stockhammer”)
1010	Perez, et al., “Poisson Image Editing,” ACM, ACM Transactions on Graphics (TOG), vol. 22, Issue 3, July 2003, pp. 313-318 (“Perez”)
1011	Wexler, et al., “Space-Time Video Completion,” In Proc. IEEE Comput. Soc. Conf. Computer Vision and Pattern Recognition, 2004, pp. 120-127 (“Wexler”)
1012	Raskar, et al., “Image Fusion for Context Enhancement and Video Surrealism,” Proceedings of the 3rd Symposium on Non-Photorealistic Animation and Rendering, 2004, pp. 85-152 (“Raskar”)

1013	Sun, et al., “Poisson Matting,” ACM Transactions on Graphics (Tog), Special Issue: Proceedings of the 2004 Siggraph Conference Session: Interacting With Images, vol. 23, Issue 3, 2004, pp. 315-321 (“Sun”)
1014	U.S. Patent No. 7,636,128 B2 to Sun, et al. (“Sun-128”)
1015	Fattal, et al., “Gradient Domain High Dynamic Range Compression,” In Proceedings of ACM SIGGRAPH, 2002, 249-256 (“Fattal”)
1016	ITU-T Rec. H.263 (01/2005) (“the H.263 standard”)
1017	M. Wada, “Selective Recovery of Video Packet Loss Using Error Concealment,” IEEE Journal on Selected Areas of Communication, vol. 7, pp. 807-814, June 1989 (“Wada”)
1018	Declaration of June Ann Munford
1019	[reserved]
1020	[reserved]
1021	ITU-T Rec. H.263 (02/1998) (“H.263-1998”)
1022	List, et al., “Adaptive Deblocking Filter,” IEEE Journal on Circuits and Systems for Video Technology, vol. 13, pp. 614-619, July 2000 (“List”)
1023	U.S. Patent No. 4,743,967 (“Takenaka”)
1024	Sun, et al., “Concealment of damaged block transform coded images using projection onto convex sets,” IEEE Trans. Image Processing, vol. 4, pp. 470–477, Apr. 1995 (Sun-2)
1025	Kwok, et al., “Multi-directional interpolation for spatial error concealment,” IEEE Trans. Consumer Electron., vol. 39, pp. 455–460, Aug. 1993 (Kwok)
1026	U.S. Patent No. 4,633,325 (“Usubuchi”)
1027	R. Graham, “Predictive Quantizing of Television Signals,” Bell Telephone Labs., 1958 (“Graham”)
1028	Third Amended Docket Control Order

LIST OF CHALLENGED CLAIM ELEMENTS

Claim 1	
[1.pre]	A moving-picture coding apparatus comprising:
[1.1.ENC]	a predictive encoder to produce and encode a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided;
[1.2.ENC]	a zone-border motion estimator to obtain a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones,
[1.3.ENC]	find a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[1.4.ENC]	and generate border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found; and
[1.5.ENC]	a zone-border motion compensator to define a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data,
[1.6.ENC]	and generate an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture,
[1.7.ENC]	wherein the predictive encoder produces the residual picture with the first predictive picture as the predictive picture and encodes the residual picture;
[1.8.ENC]	said predictive encoder including a first subtractor to produce a first residual picture from a difference between the picture to be coded and the first predictive picture;

[1.9.ENC]	an orthogonal transformer to perform orthogonal transform to the first residual picture, thus generating orthogonal-transform coefficients data;
[1.10.ENC]	a quantizer to perform quantization to the orthogonal transform coefficients data based on a specific quantization parameter, thus generating post-quantization data;
[1.11.DEC]	an inverse-quantizer to perform inverse-quantization to the postquantization data based on a specific quantization parameter, thus generating post-inverse-quantization data; and
[1.12.DEC]	an inverse-orthogonal transformer to perform inverse-orthogonal transform to the post-inverse-quantization data, thus producing a decoded residual picture.
Claim 2	
[2.1]	The moving-picture coding apparatus according to claim 1 wherein the predictive encoder further includes: a motion estimator to perform motion-vector search in the reference picture by block matching for each rectangular zone in the picture to be coded, thus generating motion vector data that indicates a matched rectangular zone in the reference picture;
[2.2]	a motion compensator to find a corresponding rectangular zone in the reference picture based on the motion-vector data, thus producing a second predictive picture;
[2.3]	a second subtractor to produce a second residual picture that is a difference between the second predictive picture and the picture to be coded; and
[2.4]	a residual determiner to compare the first predictive picture and the second predictive picture based on specific criteria, thus selecting the predictive picture having a smaller data amount.
Claim 3	
[3.pre]	The moving-picture coding apparatus according to claim 1
[3.1]	wherein the predictive encoder includes an entropy encoder to perform entropy encoding, at least, to the post-quantization data and the border motion-vector data, thus generating coded bitstreams,

[3.2]	wherein the moving picture coding apparatus comprises a multiplexer to multiplex the coded bitstreams based on the specific syntax structure.
Claim 4	
[4.pre]	A non-transitory computer readable device having stored thereon a computer program comprising a set of instructions when executed by a computer to implement a method for moving-picture coding, the program comprising:
[4.1.ENC]	a predictive encoding program code to produce and encode a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided;
[4.2.ENC]	a zone-border motion estimation program code to obtain a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones,
[4.3.ENC]	find a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[4.4.ENC]	and generate border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found; and
[4.5.ENC]	a zone-border motion compensation program code to define a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data,
[4.6.ENC]	and generate an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture,

[4.7.ENC]	wherein the predictive encoding program code produces the residual picture with the first predictive picture as the predictive picture and encodes the residual picture;
[4.8.ENC]	said predictive encoding program code including a first subtraction program code to produce a first residual picture from a difference between the picture to be coded and the first predictive picture;
[4.9.ENC]	an orthogonal transform program code to perform orthogonal transform to the first residual picture, thus generating orthogonal-transform coefficients data;
[4.10.ENC]	a quantization program code to perform quantization to the orthogonal-transform coefficients data based on a specific quantization parameter, thus generating post-quantization data;
[4.11.DEC]	an inverse-quantization program code to perform inverse-quantization to the post-quantization data based on a specific quantization parameter, thus generating post-inverse-quantization data;
[4.12.DEC]	an inverse-orthogonal transform program code to perform inverse-orthogonal transform to the post-inverse-quantization data, thus producing a decoded residual picture; and
[4.13.ENC]	an entropy coding program code to perform entropy coding, at least, to the post-quantization data and the border motion-vector data, thus generating coded bit strings,
[4.14.ENC]	wherein the moving-picture coding program further comprises multiplex program code to multiplex the coded bit strings based on a specific syntax structure, thus generating a coded bitstream.
Claim 5	
[5.1]	The moving-picture coding program code according to claim 4 wherein the predictive encoding program code further includes: a motion estimation program code to perform motion-vector search in the reference picture by block matching for each rectangular zone in the picture to be coded, thus generating motion-vector data that indicates a matched rectangular zone in the reference picture;

[5.2]	a motion compensation program code to find a corresponding rectangular zone in the reference picture based on the motion-vector data, thus producing a second predictive picture;
[5.3]	a second subtraction program code to produce a second residual picture that is a difference between the second predictive picture and the picture to be coded; and
[5.4]	a selection program code to compare the first predictive picture and the second predictive picture based on specific criteria, thus selecting the predictive picture having a smaller data amount,
[5.5]	wherein the moving-picture coding program code further comprises a coding control program code to perform control to supply the selected residual picture and a predictive picture corresponding to the residual picture to the predictive encoding program code.
Claim 6	
[6.pre]	A moving-picture decoding apparatus comprising:
[6.1.DEC]	a demultiplexer to demultiplex coded data from an input signal based on a specific syntax structure,
[6.2.ENC]	the input signal being obtained by multiplexing a coded bitstream obtained by predictive coding, border motion-vector data and post-quantization data obtained by quantization in the predictive coding,
[6.3.ENC]	the coded bitstream obtained by producing and encoding a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided,
[6.4.ENC]	obtaining a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones,

[6.5.ENC]	finding a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[6.6.ENC]	and generating the border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found, defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion vector data,
[6.7.ENC]	and generating an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing the predictive picture;
[6.8.DEC]	an entropy decoder to perform entropy decoding to the data thus demultiplexed to generate, at least, motion-vector data, the postquantization data, the border motion-vector data and parameter data required for constructing a specific syntax structure;
[6.9.DEC]	an inverse-quantizer to perform inverse-quantization to the postquantization data to generate post-quantization orthogonal transform coefficients data;
[6.10.DEC]	an inverse-orthogonal transformer to perform inverse-orthogonal transform to the post-quantization orthogonal transform coefficients data to produce a decoded residual picture of one video area;
[6.11.DEC]	a zone-border motion compensator to define a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data, and generate an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture;
[6.12.DEC]	a combiner to combine the first predictive picture and the decoded residual picture to generate a decoded moving picture signal;

[6.13.DEC]	a memory to store the decoded moving-picture signal for at least one picture as a reference picture;
[6.14.DEC]	a motion compensator to specify a corresponding rectangular zone in the reference picture based on the motion vector data, thus generating a second predictive picture;
[6.15.DEC]	a selector to select either the first predictive picture or the second predictive picture and supply the predictive picture thus selected to the combiner; and
[6.16.DEC]	a decoding controller to receive decoding control data for decoding control from the parameter data and control the selector to switch the predictive picture to be supplied to the combiner between the first and second predictive pictures according to the decoding control data.
Claim 7	
[7.pre]	A non-transitory computer readable device having stored thereon a computer program comprising a set of instructions when executed by a computer to implement a method for moving-picture decoding, the program comprising:
[7.1.DEC]	a demultiplex program code to demultiplex coded data from an input signal based on a specific syntax structure,
[7.2.ENC]	the input signal being obtained by multiplexing a coded bitstream obtained by predictive coding, border motion vector data and post-quantization data obtained by quantization in the predictive coding,
[7.3.ENC]	the coded bitstream obtained by producing and encoding a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding, and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided,

[7.4.ENC]	obtaining a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones,
[7.5.ENC]	finding a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[7.6.ENC]	and generating the border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found, defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion vector data,
[7.7.ENC]	and generating an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing the predictive picture;
[7.8.DEC]	an entropy decoding program code to perform entropy decoding to the data thus demultiplexed to generate, at least, the postquantization data, the border motion-vector data and parameter data required for constructing a specific syntax structure;
[7.9.DEC]	an inverse-quantization program code to perform inverse-quantization to the post-quantization data to generate post-quantization orthogonal transform coefficients data;
[7.10.DEC]	an inverse-orthogonal transform program code to perform inverse-orthogonal transform to the post-quantization orthogonal transform coefficients data to produce a decoded residual picture of one video area;
[7.11.DEC]	a zone-border motion compensation program code to define a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data, and generate an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture; and

[7.12.DEC]	a combine program code to combine the first predictive picture and the decoded residual picture to generate a decoded moving-picture signal, the decoded moving-picture signal for at least one picture being stored as a reference picture.
Claim 8	
[8.pre]	The moving-picture decoding program according to claim 7 wherein the entropy decoding program code further generates motion-vector data, the program further comprises:
[8.1]	a motion compensation program code to specify a corresponding rectangular zone in the reference picture based on the motion-vector data, thus generating a second predictive picture;
[8.2]	a selection program code to select either the first predictive picture or the second predictive picture and supply the predictive picture thus selected to the combiner; and
[8.3]	a decoding controlling program code to receive decoding control data for decoding control from the parameter data and control the selector to switch the predictive picture to be supplied to the combiner between the first and second predictive pictures according to the decoding control data.
Claim 9	
[9.pre]	A moving-picture coding method to be implemented in a moving-picture coding apparatus comprising the steps of:
[9.1.ENC]	producing and encoding a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided;
[9.2.ENC]	obtaining a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones

[9.3.ENC]	finding a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[9.4.ENC]	and generating border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found; defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data,
[9.5.ENC]	and generating an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture;
[9.6.ENC]	producing a first residual picture from a difference between the picture to be coded and the first predictive picture;
[9.7.ENC]	performing orthogonal transform to the first residual picture, thus generating orthogonal-transform coefficients data;
[9.8.ENC]	performing quantization to the orthogonal-transform coefficients data based on a specific quantization parameter, thus generating post-quantization data;
[9.9.DEC]	performing inverse-quantization to the post-quantization data based on a specific quantization parameter, thus generating post-inverse-quantization data; and
[9.10.DEC]	performing inverse-orthogonal transform to the post-inverse quantization data, thus producing a decoded residual picture.
Claim 10	
[10.pre]	A moving-picture decoding method to be implemented in a moving-picture coding apparatus comprising the steps of:
[10.1.DEC]	demultiplexing coded data from an input signal based on a specific syntax structure,
[10.2.ENC]	the input signal being obtained by multiplexing a coded bitstream obtained by predictive coding, border motion-vector data and post-quantization data obtained by quantization in the predictive coding,

[10.3.ENC]	the coded bitstream obtained by producing and encoding a residual picture that is a residual signal between a picture to be coded that is an input moving-picture video signal to be subjected to coding, and a predictive picture produced from a reference picture that is a local decoded video signal for each of a plurality of rectangular zones, each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided,
[10.4.ENC]	obtaining a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones,
[10.5.ENC]	finding a border, of the reference picture, having a boundary condition that matches the boundary condition, by motion-vector search in the reference picture,
[10.6.ENC]	and generating the border motion-vector data that is data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found, defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data,
[10.7.ENC]	and generating an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing the predictive picture;
[10.8.DEC]	performing entropy decoding to the data thus demultiplexed to generate, at least, the post-quantization data, the border motion-vector data and parameter data required for constructing a specific syntax structure;
[10.9.DEC]	performing inverse-quantization to the post-quantization data to generate post-quantization orthogonal transform coefficients data;
[10.10.DEC]	performing inverse-orthogonal transform to the post-quantization orthogonal transform coefficients data to produce a decoded residual picture of one video area;

[10.11.DEC]	defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion-vector data, and generate an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation, thus producing a first predictive picture;
[10.12.DEC]	combining the first predictive picture and the decoded residual picture to generate a decoded moving-picture signal; and
[10.13.DEC]	storing the decoded moving-picture signal for at least one picture as a reference picture.

I. INTRODUCTION

Google LLC (“Petitioner” or “Google”) requests *Inter Partes* Review (“IPR”) of claims 1-10 (“the Challenged Claims”) of U.S. Patent No. 8,090,025 (the “’025 patent”) (Ex.1001).

II. IDENTIFICATION OF CHALLENGES (37 C.F.R. § 42.104(b))

- **Ground 1:** Claims 2, 5, 6, and 8 are unpatentable under 35 U.S.C. § 103 over Mualla (Ex.1005), Shirani (Ex.1006), Saito (Ex.1007), and Takenaka (Ex.1023).
- **Ground 2:** Claims 1, 3, 4, 7, and 9-10 are unpatentable under 35 U.S.C. § 103 over Mualla in view of Shirani and Saito.

The ’025 patent claims priority to two foreign applications, the earliest of which was filed April 17, 2006. Ex.1001, 1:9-13. As shown below, each prior art reference that forms the basis of Grounds 1 and 2 of this Petition pre-dates the ’025 patent’s earliest proclaimed priority date (April 17, 2006; “Critical Date”), which Petitioner does not concede. June Munford provides supporting testimony in her declaration regarding the public availability and authenticity of the asserted references. Ex.1018.

Reference	§ 102(a)/(b) Publication Date
Mualla	2002
Shirani	June 2000

Saito	January 2006
Takenaka	March 1987

III. BACKGROUND

A. The '025 Patent (Ex. 1001)

The '025 patent is directed to “a moving-picture coding apparatus, method and program, and a moving-picture decoding apparatus, method and program that, without a smoothing procedure, can produce a predictive picture maintaining continuity of a video signal.” Ex.1001, 2:30-38. In addition to known video coding and decoding techniques, the '025 patent discloses a “controller” (117a and 313a) and a “switch” (102/103 and 701) for selecting between “ordinary” motion estimation/compensation and “zone-border” motion estimation/compensation. Ex.1001, 25:21-48, 26:24-48, 28:21-39, FIGS. 5 (showing encoder) and 7 (showing decoder). The '025 patent discloses that these two modes of “ordinary” and “zone-border” motion estimation/compensation are “adaptively switched based on the data amount, thus producing high-quality decoded videos with a fewer code amount overall.” Ex.1001, 28:21-39; Ex.1003, ¶¶108-115.

The “ordinary” motion estimation/compensation is performed by a “motion estimator” (502) and a “motion compensator” (503 and 702), which the '025 patent describes in its FIG. 13 explanation. Ex.1001, 10:4-39, 21:63-22:14, 28:21-39, FIGS. 5, 7, 13; Ex.1003, ¶116.

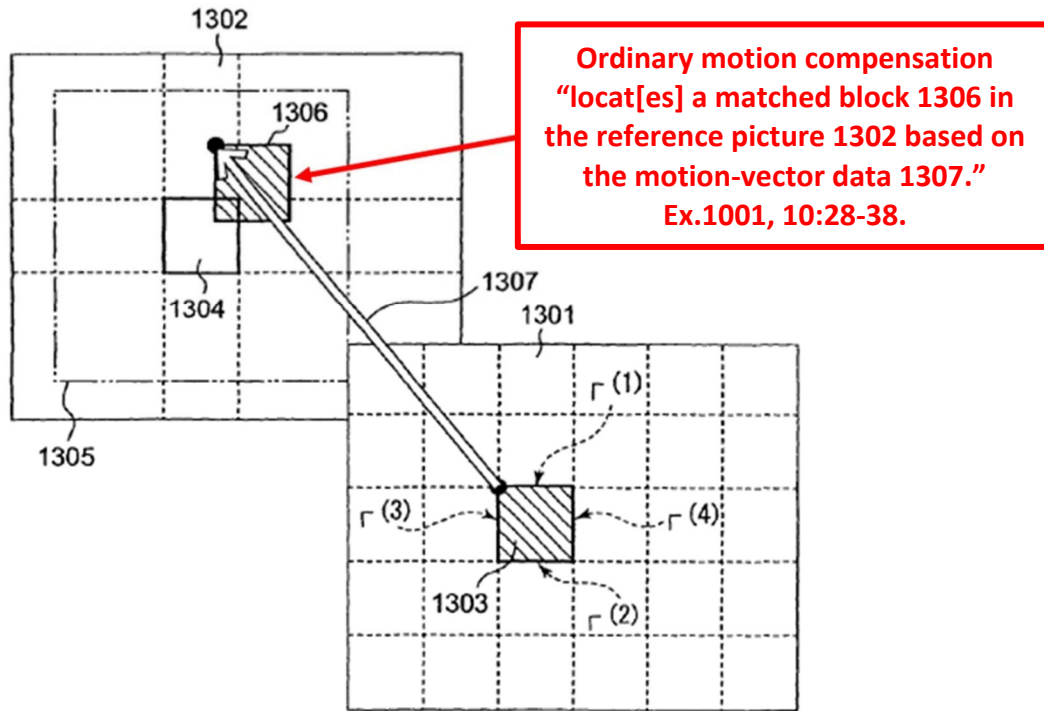


FIG. 13

Ex.1001, FIG. 13 (annotated).

The "zone-border" motion estimation/compensation includes two steps. Ex.1001, 10:39-12:20, FIGS. 14-17. In the first step, "motion-vector search is performed on block borders." Ex.1001, 10:39-45, FIGS. 14-15. Block borders are associated with a "boundary condition" which is "a gradient of a video signal on a block border." Ex.1001, 10:46-56. The zone-border motion estimator "obtains border gradient data that is a boundary condition of a block border in a picture to be coded" and then selects the "closest gradient to the gradient of an original signal on the block border." Ex.1001, 10:57-65, FIGS. 14-15 (reproduced below). In the

second step, “Poisson’s Equation is then applied based on the gradient data of the video signal thus defined to generate a predictive signal in block which is then applied to a predictive block 1606a.” Ex.1001, 11:20-34, FIGS. 16-17 (reproduced below); Ex.1003, ¶¶117-135.

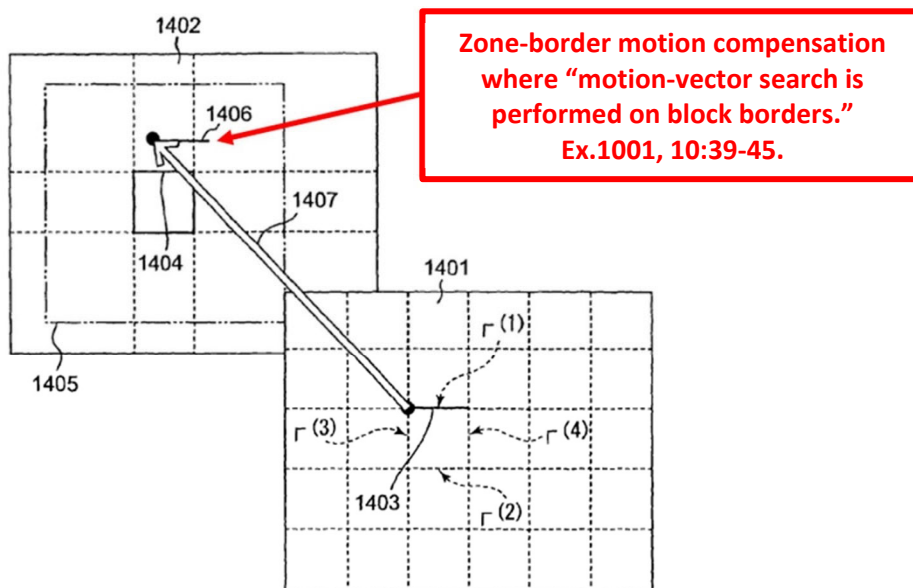


FIG. 14

Ex.1001, FIG. 14 (annotated).

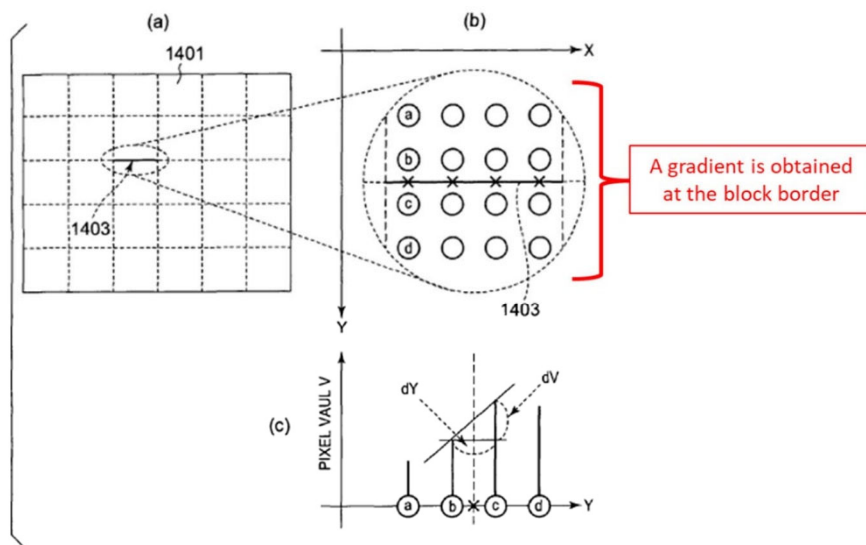


FIG. 15

Ex.1001, FIG. 15 (annotated).

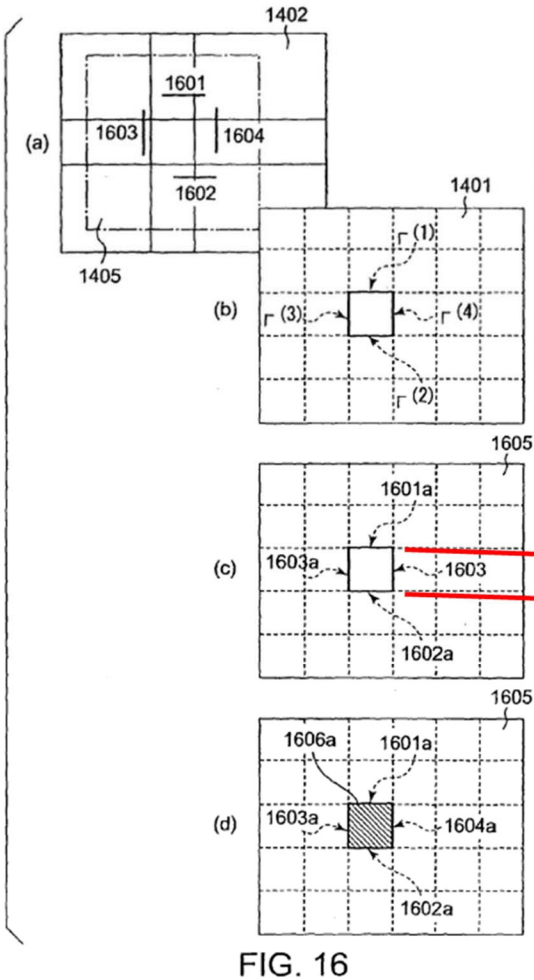


FIG. 16

Poisson's equation is applied to the gradient data to generate an "estimated signal 1706."
Ex.1001, 11:56-12:3.

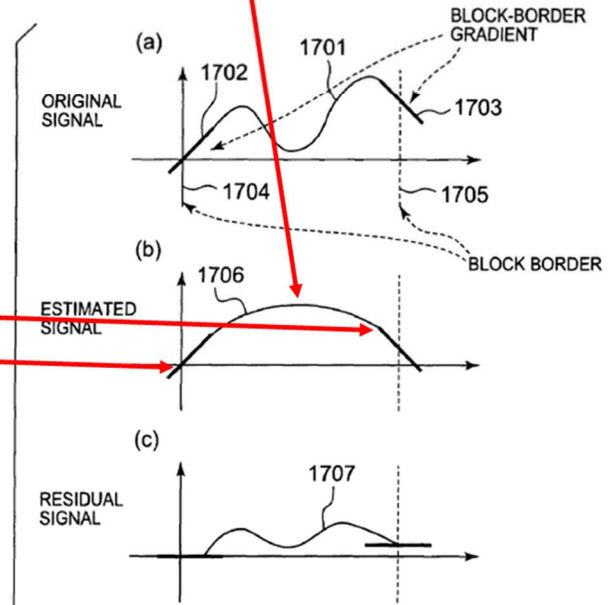


FIG. 17

Ex.1001, FIGS. 16-17 (annotated).

B. Prosecution History

The file history for the '025 patent contains no indication that the Examiner ever considered any of the prior art cited in this Petition in connection with examination of the application that issued as the '025 patent. If the art and Grounds asserted herein had been considered during examination, the '025 patent never would have been issued.

C. Person of Ordinary Skill in the Art

A person of ordinary skill in the art (“POSITA”) relating to the subject matter of the ’025 patent as of the Critical Date would have had at least a bachelor’s degree in computer science, electrical engineering, computer engineering, or a related field, with at least two years of experience in digital video processing, or a related field. Ex.1003, ¶¶105-106. Additional graduate education could substitute for professional experience, and *vice versa*. *Id.*

D. Claim Construction

The prior art relied on in this Petition disclosed the subject matter of the Challenged Claims under any reasonable construction, including their plain meaning.¹ Petitioner reserves the right to respond to any constructions offered by Patent Owner or adopted by the Board.

Petitioner notes that the Board previously addressed construction of certain claim terms in the ’025 patent in IPR2024-00374, finding it unnecessary to formally construe any claim term for purposes of institution. *See Samsung Elecs. Co., Ltd. v. Advanced Coding Techs. LLC*, IPR2024-00374, Paper 10 at 14 (PTAB Jul. 22, 2024).

¹ Petitioner reserves the right to argue alternative constructions in other proceedings, and where such a defense is available, that the claims are indefinite.

IV. GROUNDS 1-2

As demonstrated herein, the Challenged Claims recite similar limitations and overlapping subject matter. In particular, independent challenged claims 1, 4, 7, 9 and 10 recite a subset of the subject matter recited in independent claim 6.

Accordingly, Petitioner presents an analysis of claim 6's obviousness over Mualla, Shirani, Saito, and Takenaka. After Petitioner's analysis of claim 6, a table demonstrating the overlapping subject matter of the other Challenged Claims follows, showing how each Challenged Claim is rendered obvious by the combined teachings of the references cited herein. Of those remaining claims, certain Challenged Claims (1, 3, 4, 7, and 9-10) do not recite the features for which Petitioner relies upon Takenaka, and thus those claims are unpatentable under Ground 2 (Mualla, Shirani, and Saito), while Challenged Claims 2, 5, and 8 are unpatentable under Ground 1 (Mualla, Shirani, Saito, and Takenaka).

A. Overview of the Prior Art

1. Mualla (Ex.1005)

Mualla is a treatise on “fundamentals and standards of video source coding” and addresses “areas of coding efficiency, computational complexity, and error

resilience in a single volume.” Ex.1005, 9-10.² Dr. Reader provides a summary of the organization and structure of Mualla. Ex.1003, ¶¶138-173.

Given its comprehensive nature, Mualla presents different details of encoders/decoders in different figures, depending on what is being explained, but these figures are substantively cumulative. Ex.1003, ¶¶139-143. To facilitate understanding, aggregated views of the relation between different figures of Mualla referenced throughout this petition are shown in Diagrams A and B, below, which are described in more detail by Dr. Reader. Ex.1003, ¶¶139-143.

Diagram A (below) illustrates the relation between Figures 9.1 and 2.3 of Mualla. Figure 9.1 illustrates a “typical video communication system” which includes a “source encoder” designed to “compress the input video.” Ex.1005, Figure 9.1, 227-228. The source encoder “consists of a *waveform encoder* and an *entropy encoder*.” Ex.1005, 228. Further details of the source encoder are shown in Figure 2.3. The waveform encoder of Figure 9.1 “corresponds to the *mapper* and *quantizer* blocks” of Figure 2.3, whereas “the entropy encoder” of Figure 9.1 “corresponds to the *symbol encoder* block” of Figure 2.3. Ex.1003, ¶140; Ex.1005, Figures 9.1, 2.3, 228.

² Citations to Mualla are to page numbers added consistent with 37 CFR § 42.63(d)(2)(i).

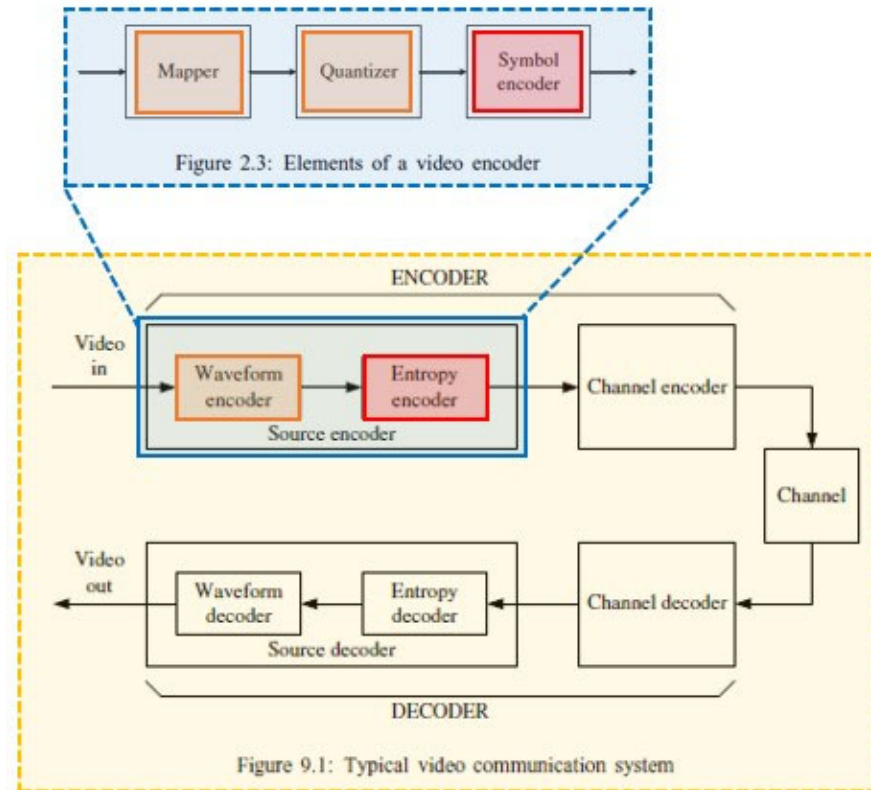


Diagram A (showing the relation between Mualla’s Figures 9.1 and 2.3)

The “mapper” in Figure 2.3 “is designed to reduce **statistical redundancy** and make the data more amenable to compression in later stages.” Ex.1005, 40; Ex.1003, ¶141. Mualla describes two types of statistical redundancy: “**spatial** (*intraframe*) and **temporal** (*interframe*) correlations between neighboring pels.” *Id.*

Spatial (*intraframe*) redundancy is reduced by “*intraframe coding*” which can be implemented by “**transform coding**.” Ex.1005, 50-51. **Temporal** (*interframe*) redundancy can be reduced by “*interframe coding*” which can be implemented by “**motion estimation (ME)**” and “**motion compensation (MC)**.”

Ex.1005, 60-61; Ex.1003 ¶142. Mualla provides further details of these techniques in Figures 1.1 and 2.9, described next.

Diagram B (below) illustrates the relation between Figure 2.3 (annotated in blue in Diagram A, above) and Figures 1.1 and 2.9 of Mualla. Figure 2.9 discloses details of transform coding (to reduce spatial redundancy) while Figure 1.1 discloses details of motion estimation and motion compensation (to reduce temporal redundancy). Figure 1.1 illustrates a “typical video codec” which includes a “motion estimation (ME) block” and a “motion compensation (MC) block” (included as part of a local “decoder” implemented at the encoder) which are designed reduce temporal redundancy. Ex.1005, 26. Figure 1.1 also illustrates a “transform encoder” block. Ex.1005, Figure 1.1. Further details of the “transform encoder” are provided in Figure 2.9, which includes a “forward transform” designed “to decorrelate the pels³ of the input block” (reduce spatial redundancy). Ex.1005, 51-52; Ex.1003, ¶143.

³ “Pels” is another term for picture elements or pixels.

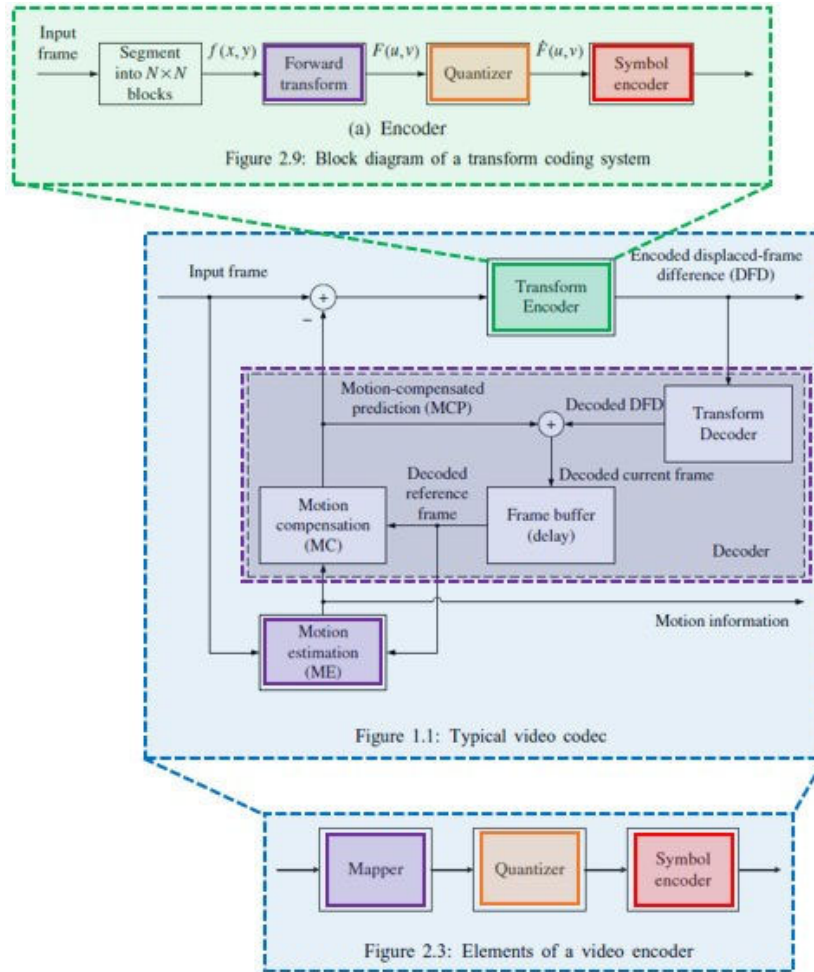


Diagram B (showing the relation between Mualla's Figures 2.3, 1.1, and 2.9)

Mualla further describes the “picture coding structure” of the “H.263 standard” in which “input video consists of a sequence of pictures (or frames)” and “[e]ach picture is divided into *groups of blocks* (GOBs)” where “[e]ach GOB is divided into *macroblocks* (MBs)” and each MB “consists of 16×16 samples of” (16×16 pixel elements or “pels”). Ex.1005, 71-73, Figure 3.2; Ex.1003, ¶144.

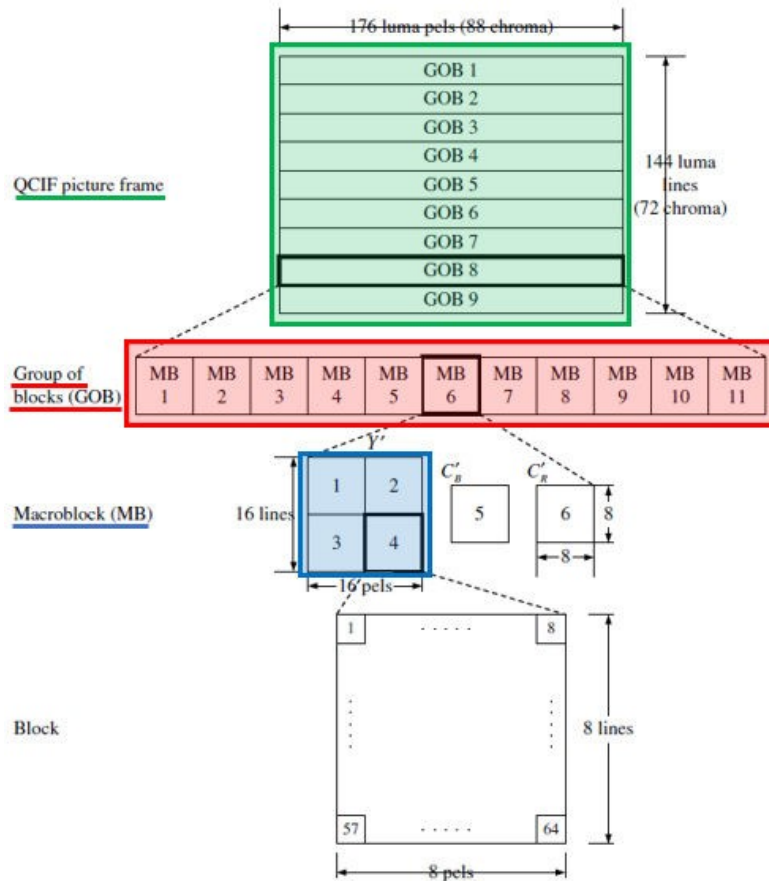


Figure 3.2: H.263 picture structure for a QCIF frame

Ex.1005, Figure 3.2 (annotated and color added)

When a picture frame of input video is encoded, “GOBs are coded from top to bottom in increasing number” and “[w]ithin each GOB, the MBs are coded from left to right (and from top to bottom if the GOB contains more than one row of MBs) in increasing number.” Ex.1005, 71, Figure 3.2; Ex.1003, ¶145.

To encode an “input frame” (the “current frame”) of video, the encoder:

(i) uses “motion estimation (ME)” and “motion compensation (MC)” to generate “a prediction of the current frame” and

(ii) subtracts the “prediction” of the current frame from the current frame “to produce an error signal known as the displaced-frame difference (DFD)” which is then encoded using the “transform encoder.” Ex.1005, 26-27, Figure 1.1; Ex.1003, ¶146.

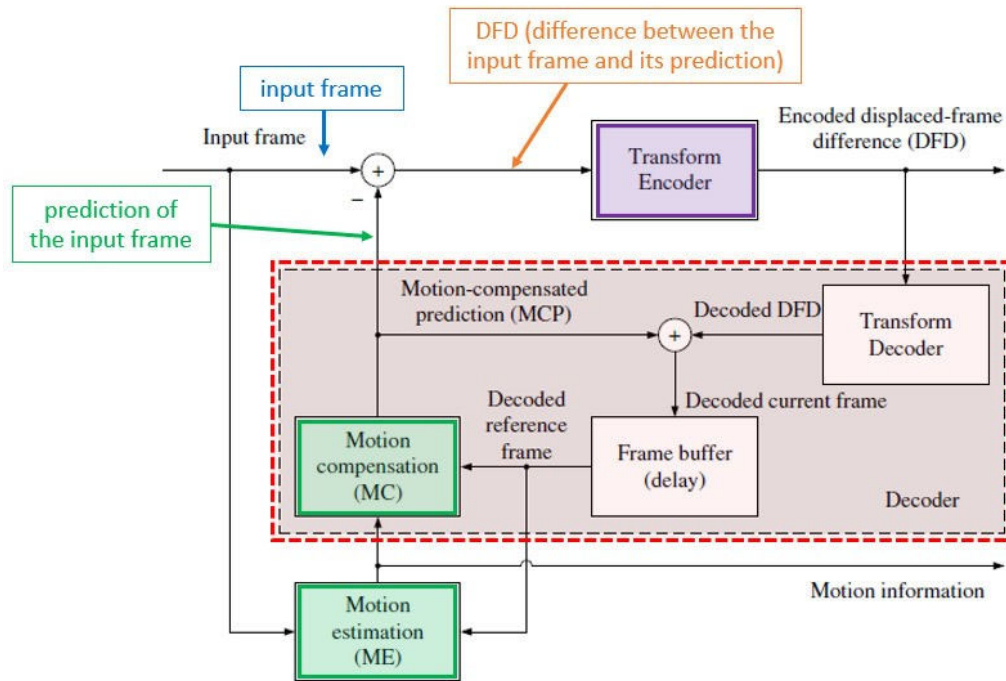


Figure 1.1: Typical video codec

Ex.1005, Figure 1.1 (annotated and color added)

Mualla provides further details of these two operations, summarized below.

Ex.1003, ¶¶147-157.

(i) Generating the “prediction” of the current frame

As shown in Figure 1.1 (below), for a given input frame, a “prediction” of the current frame is generated by a “motion compensation (MC) block” which uses “motion information” estimated by a “motion estimation (ME) block.” Ex.1005, 26, *see also* 61, Figure 1.1; Ex.1003, ¶148.

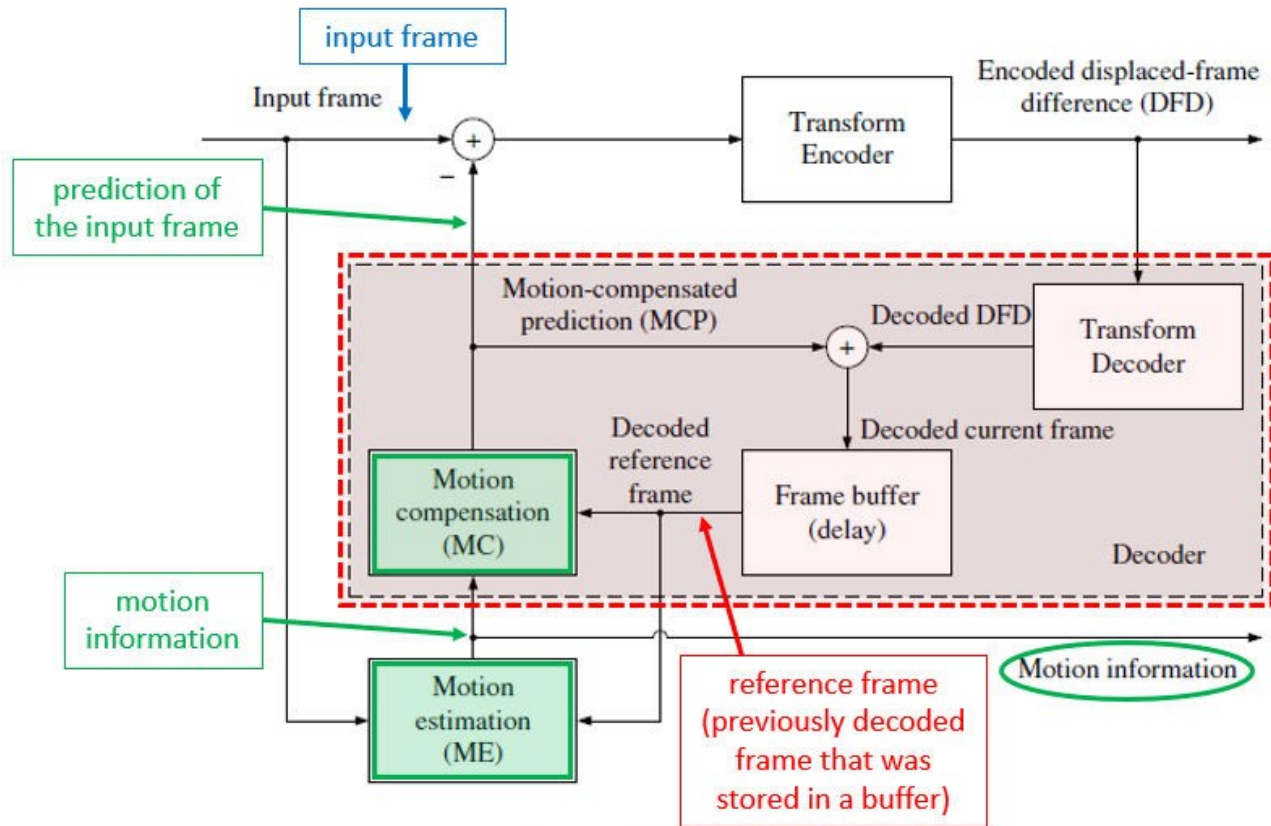


Figure 1.1: Typical video codec

Ex.1005, Figure 1.1 (annotated and color added)

The “motion information” is an estimate of “movement that occurred between the current frame and a reference frame (usually a previously decoded frame that was stored in a frame buffer).” Ex.1005, 26, *see also* 61. Mualla explains that “[t]he most commonly used ME method is the block-matching motion estimation (BMME) algorithm” in which the “motion of each block” is estimated “by searching for the best-match block in the reference frame.” Ex.1005, 62. The motion of the current block is “represented by a motion vector, which is the displacement between the block and its best-match block in the reference frame.” *Id.*, Figure 2.16; Ex.1003, ¶149.

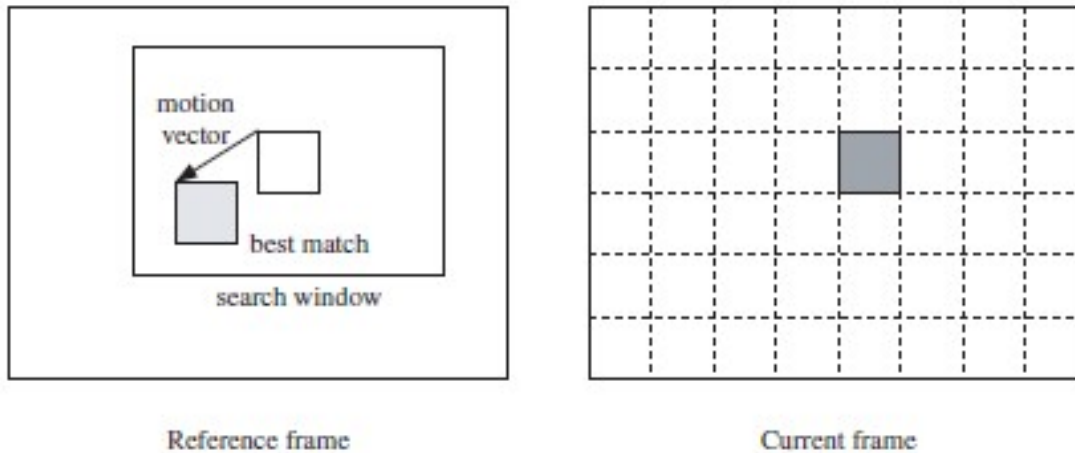


Figure 2.16: Block-matching motion estimation

Ex.1005, Figure 2.16.

After the motion estimation (ME) process has estimated a motion vector, the encoder performs “motion compensation (MC)” in which the motion vector is utilized “to move the contents of the reference frame to provide a better prediction of the current frame.” Ex.1005, 61; Ex.1003 ¶150. The resulting prediction (annotated in green in Figure 1.1, above) is “the motion-compensated prediction (MCP) or the displaced-frame (DF).” *Id.*

The video encoder implements motion compensation (MC) by using a local “decoder” that locally decodes a frame, stores the decoded frame in the local frame buffer, and generates the prediction of the current frame based on the previously decoded and stored frame. Ex.1005, 26, Figure 1.1; Ex.1003, ¶151.

In addition to supplying the estimated motion vector to the motion compensation (MC) block, the motion vector is also encoded and output from the

encoder (see “motion information” circled in the lower-right-hand corner of Figure 1.1, above). Ex.1005, Figure 1.1; Ex.1003, ¶152. For each macroblock, “[t]he estimated motion vector $MV = (MV_x, MV_y)$ is predictively coded” which means that “the motion vector difference $MVD = MV - MVP$ is encoded instead of the motion vector itself.” Ex.1005, 74. The motion vector difference (MVD) is then “entropy coded” and the encoded MVD is output from the encoder as encoded motion information. Ex.1005, 74; Ex.1003, ¶152.

(ii) Encoding the DFD

After a prediction of the current frame is generated in step (i), “the prediction is then subtracted from the current frame to produce an error signal known as the displaced-frame difference (DFD)” (shown below) which is encoded using a “transform encoder” to produce an “encoded...DFD” which is output from the encoder. Ex.1005, Figure 1.1; Ex.1003, ¶153.

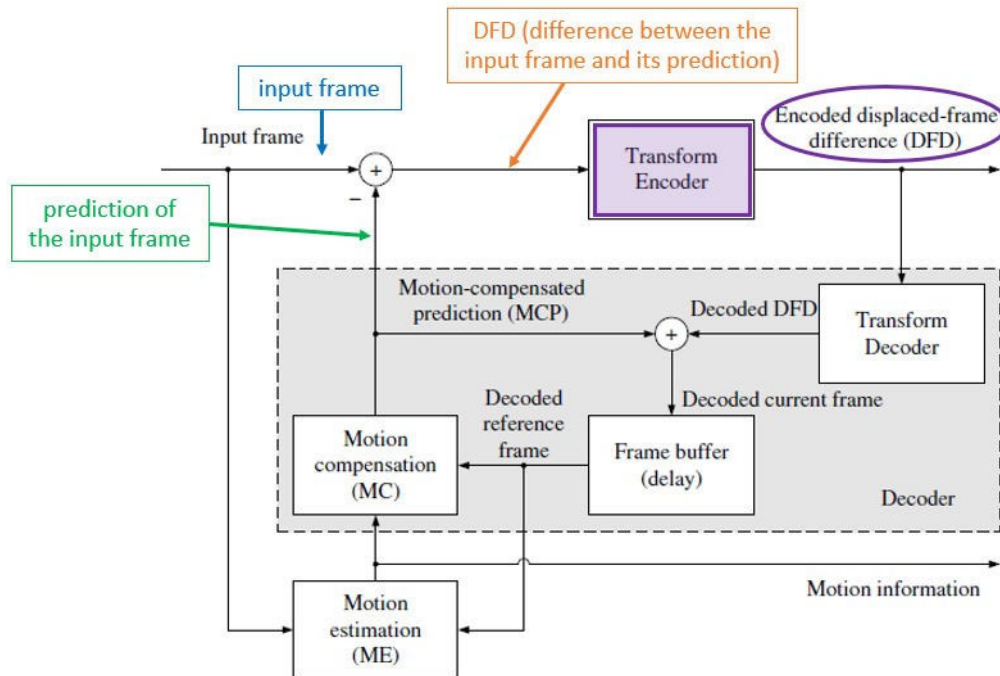


Figure 1.1: Typical video codec

Ex.1005, Figure 1.1 (annotated and color added)

Mualla describes details of the “transform encoder” in Figure 2.9 (below), in which “the input frame is first **segmented** into $N \times N$ blocks” and then “a unitary 16 space-frequency **transform** is applied to each block to produce an $N \times N$ block of transform (spectral) coefficients that are then suitably **quantized** and **coded**.”

Ex.1005, 51-52; Ex.1003, ¶154.

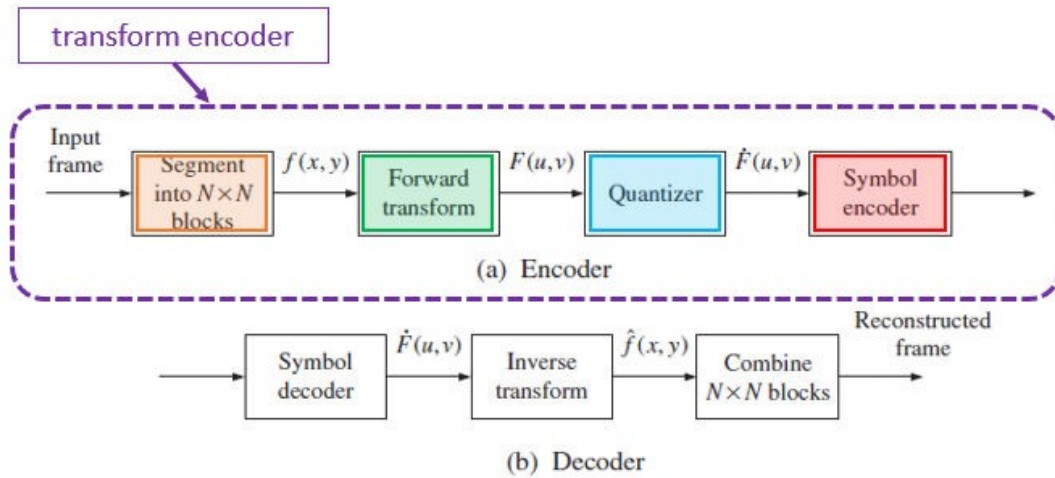


Figure 2.9: Block diagram of a transform coding system

Ex.1005, Figure 2.9 (annotated and color added)

Thus, the **encoder** outputs both the encoded DFD (circled in purple in Figure 1.1) and the encoded motion information (circled in green in Figure 1.1). Ex.1005, 26, 52, Figure 1.1; Ex.1003, ¶155.

Mualla explains that “[a]t the **decoder**, the reverse operations are performed to obtain the output video.” Ex.1005, 228. Figure 9.1 (below); Ex.1003, ¶156.

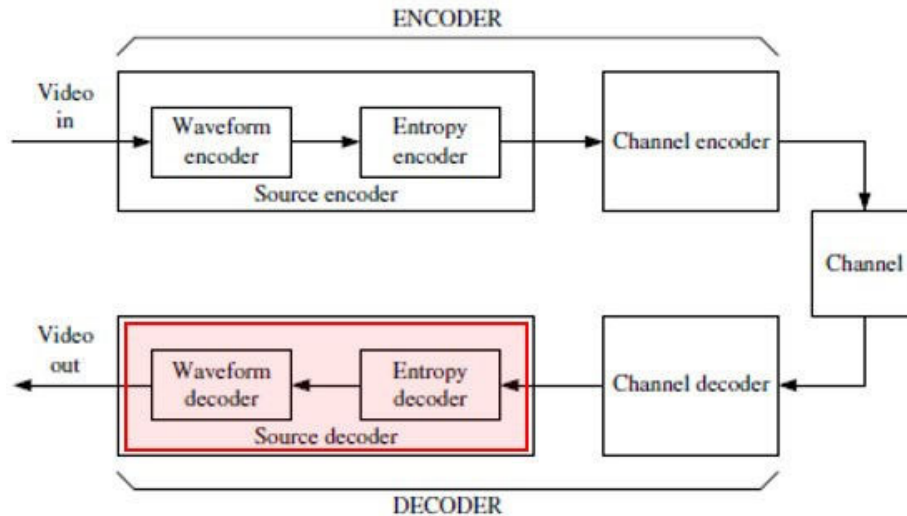


Figure 9.1: Typical video communication system

Ex.1005, Figure 9.1 (color added)

These reverse operations in the “decoding process” include (i) “motion vector decoding” in which “for each transmitted inter MB, the decoder calculates the same motion vector predictor MVP used at the encoder and adds it to the decoded motion vector difference MVD to obtain the decoded motion vector MV,” (ii) “motion compensation” in which “[t]he decoded motion vector is used to compensate” the macroblock, (iii) “inverse quantization” for “reconstructing an approximation of the original coefficient,” (iv) “inverse transform” in which “[t]he reconstructed block of coefficients is processed by a separable 2-D 8x8 inverse DCT,” and (v) “reconstruction of blocks” in which “the reconstructed block is formed by summing the motion-compensated prediction and the result of the inverse DCT.” Ex.1005, 76; Ex.1003, ¶157.

Error Resilience Techniques

In addition to the conventional video coding and decoding operations discussed above, Mualla further describes techniques for “error resilience” in video coding, including “error detection and error control techniques” applicable to the H.263 standard. Ex.1005, 11, 227-251. Mualla explains that “when transmitted over a mobile channel, compressed video can suffer severe degradation and the use of error-resilience techniques is vital.” Ex.1005, 25; Ex.1003, ¶158.

To implement error resilience, Mualla describes a system in which “[t]he decoder was modified to perform error detection by detecting syntax and semantic violations.” Ex.1005, 265. In particular, “[w]hen an error is detected, the decoding process is stopped” and “[a]ll macroblocks between the point where the error was detected and the synchronization point are marked as damaged macroblocks.” *Id.* The damaged macroblocks are recovered by using an error control technique called “**hybrid** error concealment” in which “**temporal** concealment is first used to get an initial estimate of the damaged block” and “this initial estimate is then refined using **spatial** concealment.” Ex.1005, 242-246, Figures 9.7, 9.8 (reproduced below); Ex.1003, ¶159.

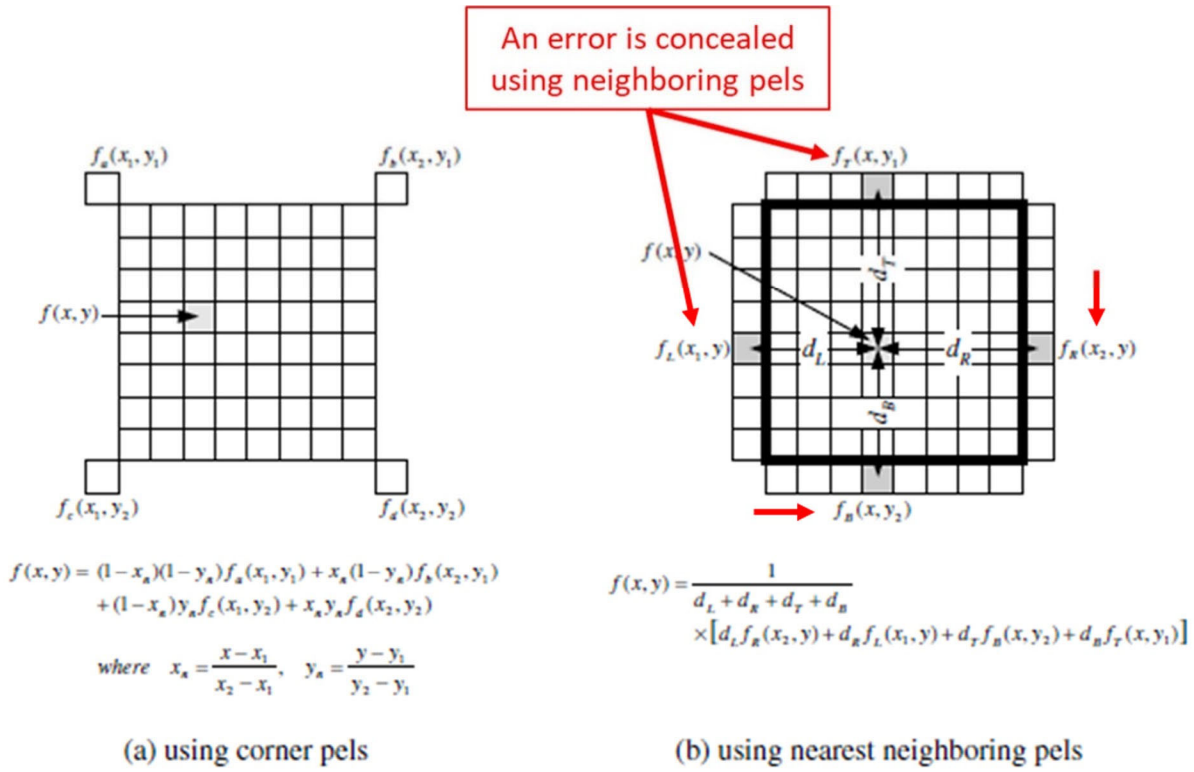


Figure 9.7: Error concealment using spatial interpolation

Ex.1005, FIG. 9.7 (annotated).

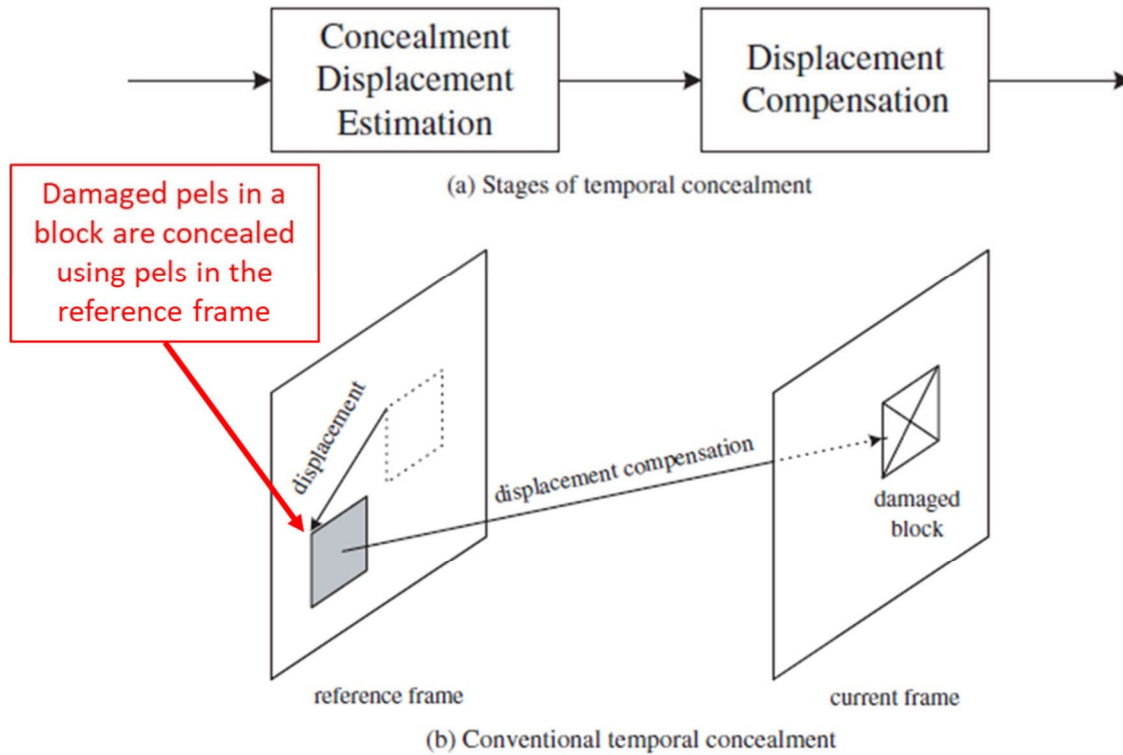


Figure 9.8: Temporal error concealment

Ex.1005, FIG. 9.8 (annotated).

Furthermore, Mualla describes that error concealment can be performed at **both** the **encoder** and the **decoder**. Ex.1005, 246-250. For example, “the encoder and decoder cooperate to minimize the effects of transmission errors” and in particular “coding efficiency can also be improved by **performing error concealment** in the **encoder** so that both encoder and decoder use the same reference frames for prediction.” Ex.1005, 249. To implement error concealment at the encoder, Mualla describes “**interactive techniques**” in which “the decoder uses a feedback channel to inform the encoder about which parts of the transmitted video have been received in error” and “[b]ased on this feedback information, the

encoder adjusts its operation to combat the effects of such errors.” Ex.1005, 224.

Mualla explains that “[w]hen feedback information is received, the encoder can reconstruct the error propagation process” so that “the encoder can track the error propagation.” Ex.1005, 247; Ex.1003, ¶160. Mualla also cites to numerous corroborating references, such as Girod (Ex.1008) and Wada (Ex.1017), for additional details of implementing “interactive” error concealment at both the encoder and decoder. Ex.1005, 246-250. Dr. Reader provides a summary of these corroborating references cited in Mualla. Ex.1003, ¶161.

Therefore, in view of the teachings of Mualla (and the references cited therein which reflect knowledge in the art at the time of Mualla’s teachings), a POSITA would have understood or found obvious that (i) an H.263-based **decoder** performs *error detection* to detect errors and *error concealment* to recover from those errors, and (ii) an H.263-based **encoder** receives feedback from the decoder regarding errors that were detected, and performs the same *error concealment* to replicate those same errors in its local decoder so as to use the same reference frames for error prediction. Ex.1003, ¶¶162-168.

To summarize, Mualla provides a self-contained teaching of video coding systems, including feedback-based error tracking in which (i) an H.263-based **decoder** performs *error detection* to detect errors and *error concealment* to recover from those errors, and (ii) an H.263-based **encoder** receives feedback from the

decoder regarding errors that were detected, and performs the same *error concealment* to replicate those same errors in its local decoder so as to use the same reference frames for error prediction. Mualla refers to (i) as a “post-processing” technique and refers to (ii) as an “interactive” technique, and Mualla explains that a “combination” of these techniques can be employed in practical systems. Ex.1005, 251; Ex.1003, ¶¶162-173.

These teachings of Mualla are corroborated by Girod and Wada (both cited in Mualla) which provides additional details and diagrams for the same concepts disclosed in Mualla. Ex.1003, ¶¶162-173.

Therefore, in view of the teachings of Mualla (as corroborated by Girod and Wada), a POSITA would have understood or found obvious that an error resilience technique (such as the “hybrid error concealment” technique described in Mualla) can be implemented at both the encoder and the decoder in a feedback-based video coding system. Ex.1003, ¶¶162-173.

2. Shirani (Ex.1006)

Shirani is cited in Mualla (as reference [199]) for teaching details of the “**hybrid** error concealment” technique that was described in Mualla. Ex.1005, 246. Specifically, Shirani teaches a “**two-stage** error-concealment method for block-based compressed video which was transmitted in an error-prone environment” in an H.263 video coding framework. Ex.1006, Abstract; 4.

Shirani's "**first** stage" "obtain[s] an initial estimate of the missing block using information from the previous frame," which Shirani refers to as "**temporal**" concealment (consistent with Mualla). Ex.1005, 243-245; Ex.1006, 1-2. In Shirani's "**second** stage," "the information around each missing block is used to refine the initial estimate," which Shirani refers to as "**spatial**" concealment (consistent with Mualla). Ex.1005, 242-243; Ex.1006, 1-3; Ex.1003, ¶174.

The **first stage** (temporal error concealment) utilizes motion vectors of "subblocks adjacent to the missing block" (subblocks "adjacent above and left" as well as "subblocks ... to the right and below of the missing block") to derive an "initial estimate" of the missing block that has the smallest sum of "border errors." Ex.1006, 3. Shirani illustrates an example of "subblocks adjacent to the missing block, i.e., $u1$, $u2$, $l1$, and $l2$ in Fig. 1(a)." Ex.1006, 3, FIG. 1(a); Ex.1003, ¶175.

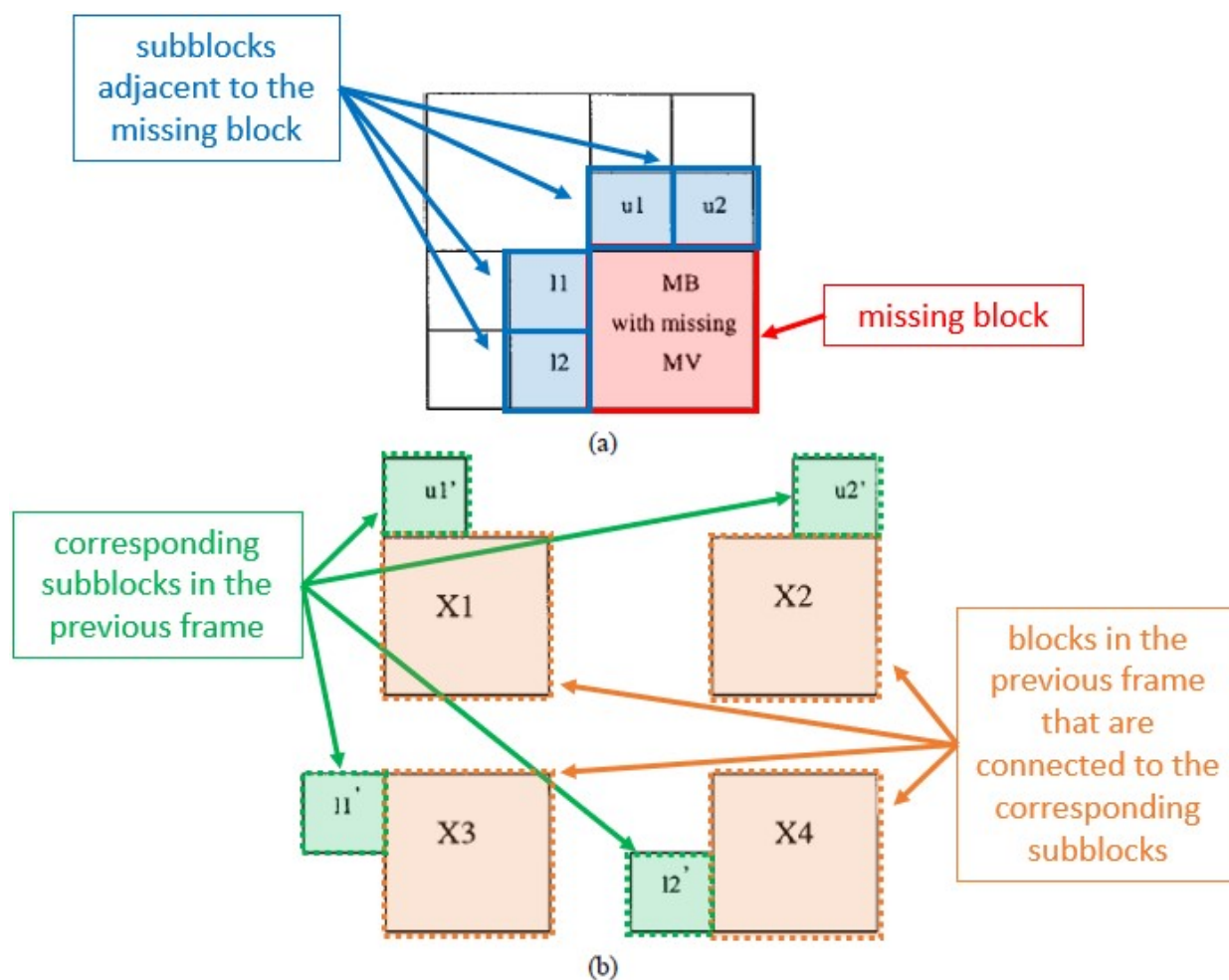


Fig. 1. (a) The four subblocks, (b) their corresponding subblocks in the previous frame, and the blocks connected to them.

Ex.1006, FIG. 1(a) (annotated and color added).

Once these adjacent subblocks ($u1$, $u2$, $l1$, $l2$) are identified, “for each of these subblocks, a corresponding subblock in the previous frame is determined” (denoted $u1'$, $u2'$, $l1'$, and $l2'$ at the bottom of FIG. 1(a) of Shirani). Ex.1006, 3. Then, “four blocks, namely X1, X2, X3, and X4, which are connected to $u1'$, $u2'$, $l1'$, and $l2'$ (respectively), are determined” and “a block from the above four blocks [(X1, X2, X3, and X4)] that minimizes the squared sums of border errors, between

the estimated block and its adjacent above and left blocks, is selected.” Ex.1006, 3; Ex.1003, ¶176. This is accomplished by defining “border error[s]” in “terms of pixels,” including an error for the “bottom line pixels of the block to the top of the missing block” and an error for the “right column pixels of the block to the left of the missing block” and applying the same technique to define border errors for pixels adjacent to “the subblocks ... to the right and below of the missing block.” *Id.*

To estimate the motion of the adjacent subblocks ($u1$, $u2$, $l1$, $l2$), Shirani discloses that “the search for displacement of each of the subblocks is restricted to a set of candidate motion vectors” which can include “the motion vectors of available neighboring blocks, the median of the motion vectors of available neighboring blocks, the average of the motion vectors of available neighboring blocks.” Ex.1006, 3; Ex.1003, ¶177. Furthermore, although the above description focuses on adjacent subblocks above and to the left of the missing block, Shirani also discloses that “the same method can be extended to include the subblocks of the blocks to the right and below of the missing block.” Ex.1006, 3.

For the **second stage** (spatial error concealment), Shirani describes various techniques that can “refine the initial estimate” (obtained in the first stage, above) by utilizing “information around each missing block.” Ex.1006, 1, 3. As examples of “spatial error concealment,” Shirani describes techniques in which neighboring

pixels around the missing block are used to “refine the initial estimate” that was obtained in the first stage. Ex.1006, 1 (“each pixel in a damaged block is interpolated from the corresponding pixels in its four neighboring blocks”), 3 (“[i]n the second stage of our proposed error-concealment method, the information around each missing block is used to refine the initial estimate”); Ex.1003, ¶178.

3. Combination of Mualla and Shirani

As discussed above, Mualla describes “hybrid error concealment” for “error resilience” in video coding that implements both temporal and spatial concealment. Ex.1005, 241-246. Mualla specifically cites Shirani for details of this hybrid error concealment technique. Ex.1005, 241-246. Given Mualla’s explicit reference to Shirani, a POSITA would have naturally looked to Shirani and its “two-stage error concealment” technique for further details of how to implement “hybrid error concealment” as Mualla described. Ex.1005, 246; Ex.1006, 1-6; Ex.1003, ¶183. A POSITA implementing the teachings of Shirani with Mualla’s existing disclosure would have created the Mualla-Shirani combination. Ex.1003, ¶183.

In particular, the hybrid error concealment techniques of the Mualla-Shirani combination perform error detection to detect whether errors exist in a frame (also referred to as a picture). Ex.1005, 241-246; Ex.1006, 2-4. If errors are detected in a frame/picture, a two-stage motion-compensated error concealment algorithm is applied to perform both temporal and spatial concealment to conceal the errors and

generate a predicted frame. Ex.1005, 242-246; Ex.1006, 2-4. Otherwise, if no errors are detected, conventional motion compensation is applied. *Id.* Specifically, as discussed above, the two-stage (hybrid) error concealment algorithm of the Mualla-Shirani combination implements a “first stage” which applies “temporal” concealment and a “second stage” which applies “spatial” concealment. Ex.1005, 242-245; Ex.1006, 1-3; Ex.1003, ¶184.

Furthermore, in the Mualla-Shirani combination, and in particular in Shirani’s “two-stage error concealment” technique, Shirani describes various alternatives for the “second stage” (in which “spatial” concealment is performed). As discussed above, as an example of the second stage, Shirani describes a spatial concealment technique in which “each pixel in a damaged block is interpolated from the corresponding pixels in its four neighboring blocks such that the total squared border error is minimized.” Ex.1006, 1; Ex.1003, ¶185.

Additionally, as discussed above, the two-stage hybrid error concealment is implemented in both the encoder and decoder (“interactive error concealment”), as Mualla teaches, so that the encoder and decoder can “cooperate to minimize the effects of transmission errors” and in particular so that “coding efficiency can also be improved by performing error concealment in the encoder so that both encoder and decoder use the same reference frames for prediction.” Ex.1005, 246-250; Ex.1003, ¶186.

As explained below in more detail, combining Mualla and Shirani would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results; and the (2) use of known techniques to improve similar devices in the same way. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415-421, 82 USPQ2d 1385, 1395-97 (2007); MPEP 2143; Ex.1003, ¶187. Indeed, Mualla describes a similar hybrid error concealment technique as Shirani, and the combination would have simply involved adding Shirani's teachings into the compatible teachings of Mualla, thus, for at least this reason, making the Mualla-Shirani combination predictable and foreseeable with a reasonable expectation of success. Ex.1005, 246; Ex.1006, 1-6; Ex.1003, ¶187. Moreover, because Mualla explicitly cites to Shirani's teachings when discussing its own techniques of hybrid error concealment, the use of Shirani's techniques in Mualla would have improved a similar device in the same way. Ex.1005, 246. Additionally, Mualla and Shirani implement their teachings in similar video coding standards (e.g., H.263). Ex.1005, 71-94, 231; Ex.1006, 4-5; Ex.1003, ¶187.

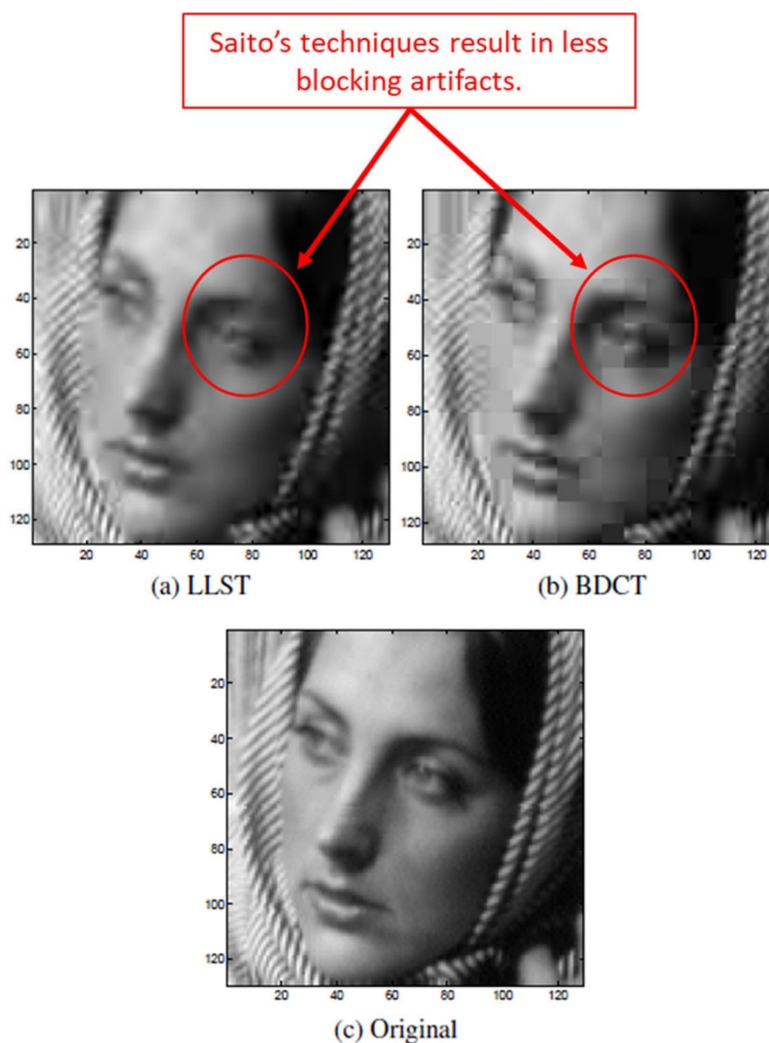
4. Saito (Ex.1007)

Saito discloses "image approximation" using techniques based on solving Poisson's equation. Ex.1007, 6, 27; Ex.1003, ¶179. Saito expresses a goal of improving "the perceptual quality of these approximations" and a desire for a

“perceptually correct metric” that can improve the “visual quality” of approximated images. Ex.1007, 18; Ex.1003, ¶179. Saito also discusses the relevance of “psychophysical experiments” in selecting appropriate image approximation techniques that improve “visual fidelity” of reconstructed images. Ex.1007, 24; Ex.1003, ¶179.

To achieve these goals, Saito discloses a “polyharmonic local sine transform” (PHLST) and a “polyharmonic local cosine transform” (PHLCT) that both solve Poisson’s equation and are “effective for image approximation.” Ex.1007, 6, 27; Ex.1003, ¶180. Saito explains that “PHLST can represent smooth regions of an image very efficiently” and therefore “image approximation and compression are immediate and important applications of PHLST.” Ex.1007, 8-9. Saito further discloses “extensions and applications of PHLST” which include a version called “polyharmonic local cosine transform (PHLCT).” Ex.1007, 26-27. The PHLCT technique can “solve the ... Poisson equation with the Neumann boundary condition.” *Id.* The PHLCT utilizes “the first order normal derivative of data at the boundary.” *Id.* In particular, the solution to Poisson’s equation has “the same normal derivative at the boundary with the original function” by ensuring that “the residual function v_j ... clearly satisfies $\partial v_j / \partial v = 0$ on $\partial \Omega_j$.” Ex.1007, 27. Thus, Saito’s PHLCT technique ensures that the derivative of the estimated signal at a boundary of a

block is equal to the derivative of the signal to be estimated at the boundary, ensuring smoothness of the estimated image across block boundaries with less “blocking artifacts.” Ex.1003, ¶180; Ex.1007, 27; *see also* 5, 18, Abstract, FIG 11.



Ex.1007, FIG. 11 (cropped and annotated).

5. Combination of Mualla-Shirani and Saito

As discussed above, the Mualla-Shirani combination implements a two-stage (hybrid) error concealment technique. The second stage applies “spatial error-

concealment” to “restore the missing blocks using the information in the current frame.” Ex.1006, 1; *see also* Ex.1005, 242. Both Mualla and Shirani disclose that “**spatial**” concealment can be implemented using various techniques from the field of **conventional image-processing**, each with benefits and tradeoffs. Ex.1005, 242-243; Ex.1006, 1; Ex.1003, ¶188. Indeed, as Dr. Reader confirms, it was well-known to use image-processing techniques to perform spatial concealment within an individual frame of a video. Ex.1003, ¶189 (citing Ex.1011, as disclosing a hybrid-type “space-time video completion” technique for “[c]orrection of missing/corrupted video frames” in which “classical spatial image completion and synthesis” is implemented for “spatial” correction within an individual video frame).

Furthermore, a POSITA would have understood or found obvious that image processing can be performed by techniques that **solve Poisson’s equation**.

Ex.1003, ¶190. As Dr. Reader explains, numerous research papers and patents had, several years before the Critical Date, described techniques for image editing and image reconstruction that involved solving Poisson’s equation. Ex.1003, ¶190 (citing Perez, Ex.1010, Raskar, Ex.1012, Sun/Sun-128, Ex.1013 and Ex.1014, and Fattal, Ex.1015).

As Dr. Reader explains, a POSITA would have understood or found obvious that image processing techniques involving Poisson’s equation are particularly

effective in addressing the same challenges and achieving the same goals emphasized by Mualla and Shirani for error concealment applications. Ex.1003, ¶191. For example, as Dr. Reader explains, Mualla repeatedly emphasizes a fundamental importance of considering the “human visual system (HVS)” in viewing the “subject quality” of video codec techniques, and in particular for error concealment. Ex.1005, 33, 36, 40, 52, 55, 59, 242, 251, 265, 268, Figures 10.8 and 10.10; Ex.1003, ¶191. Like Mualla, Shirani also stresses the fundamental importance of considering the “limitations of the **human visual system**” when designing an error concealment technique. Ex.1006, 1, 4; Ex.1003, ¶191.

In view of Mualla and Shirani emphasizing a fundamental challenge and goal of improving the “subjective quality” of error concealment for the “human visual system,” a POSITA would have understood an image approximation technique based on **solving Poisson’s equation** to be an effective way to achieve this goal. Ex.1003, ¶192. As Dr. Reader explains, “the Poisson equation has been used extensively in field of image processing and computer vision due to the specific benefits it provides for improving the perceptual quality of reconstructing images.” Ex.1003, ¶192 (citing Ex.1010).

Therefore, as discussed above: (i) Mualla and Shirani both recognize that **image processing** techniques can be utilized to implement **second-stage spatial** error concealment in the Mualla-Shirani two-stage hybrid error concealment

technique for video frames, (ii) image processing techniques that involve **solving Poisson's equation** were well-known to provide specific benefits for improving the perceptual quality of reconstructed images for the human visual system, thus addressing a fundamental objective of Mualla and Shirani's error concealment.

Ex.1003, ¶193. As such, a POSITA would have understood or found obvious that image-processing techniques based on solving Poisson's equation would have been naturally applicable for implementing the second-stage "spatial" concealment of the Mualla-Shirani two-stage hybrid error concealment algorithm for video frames.

Ex.1003, ¶193. As Dr. Reader further explains, literature in video coding spatial error concealment and literature in image processing by solving Poisson's equation both cited to similar techniques. Ex.1003, ¶193.

Saito provides details of implementing solutions to Poisson's equation for image approximation. Ex.1003, ¶194. A POSITA would have also understood or found obvious that Saito, which was published in 2006 (over 4 years after Mualla and Shirani), was a natural advancement of applying Poisson's equation to improve the visual quality of image approximation and represented a state-of-the-art solution at the time of the Critical Date. Ex.1003, ¶194. Like Mualla and Shirani, Saito expresses an overarching goal of improving "the perceptual quality of these approximations" and a desire for a "perceptually correct metric" that can improve the "visual quality" of approximated images. Ex.1007, 18; Ex.1003, ¶194. Saito

also discusses the relevance of “psychophysical experiments” in selecting appropriate image approximation techniques that improve “visual fidelity” of reconstructed images. Ex.1007, 24; Ex.1003, ¶194.

As discussed above, Saito discloses a “polyharmonic local sine transform” (PHLST) and a “polyharmonic local cosine transform” (PHLCT) that solve Poisson’s equation and are “effective for image approximation.” Ex.1007, 6, 27; Ex.1003, ¶195. In particular, Saito explains that its PHLCT technique “shall solve the [] Poisson equation with the Neumann boundary condition.” Ex.1007, 27; Ex.1003, ¶195.

Furthermore, as discussed above, a POSITA would have recognized or found obvious that Mualla, Shirani, and Saito all strive to achieve similar goals in spatial image approximation, and accordingly a POSITA would have had a reasonable expectation of success in combining Saito’s techniques into Mualla-Shirani. As Dr. Reader explains, both Shirani and Saito strive to improve reconstruction of high-frequency edges and discontinuities in an image. Ex.1003, ¶196 (citing Ex.1006, 5 and 7, and Ex.1007, 1, 2, and 4).

As discussed above, Saito’s PHLCT technique ensures that the derivative of the estimated signal at a boundary of a block is equal to the derivative of the signal to be estimated at the boundary. Ex.1007, 27; *see also* 5, Abstract; *see supra* §IV.4. A POSITA would have understood or found obvious that this is consistent

with Shirani's "second stage" technique generating a "refined initial estimate" that has border pixel values with a "border error" that is equal to the "border error" of the "initial estimate" that was generated in Shirani's "first stage." Ex.1006, 1, 3; Ex.1007, 27; *see also* 5, Abstract; Ex.1003, ¶197. As Dr. Reader explains, Shirani's "border error" is equivalent to Saito's "derivative at the boundary" (which is equivalent to the '025 patent's "gradient" value at a boundary). Ex.1003, ¶197. Furthermore, a POSITA would have understood or found obvious that because Saito's PHLCT technique can achieve a faster "decay rate" for the error signal at the boundaries of blocks, this is consistent with Shirani's goal of "improv[ing] the video reconstruction quality" by ensuring that "image continuity inside the block and across its boundaries is preserved." Ex.1007, 27; Ex.1006, Abstract, 3.

As explained below in more detail, combining Mualla-Shirani and Saito would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results; and the (2) use of known techniques to improve similar devices in the same way. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415-421, 82 USPQ2d 1385, 1395-97 (2007); MPEP 2143; Ex.1003, ¶198.

In the resulting combination of Mualla, Shirani, and Saito, if no errors are detected in a frame of video, then conventional motion compensation is applied to

the frame “to provide a prediction of the current frame,” as taught by Mualla and Shirani. Ex.1005, 27, Figure 1.1, 115-116; Ex.1006, Abstract 2-3. Otherwise, if an error is detected in one or more blocks of a frame, then a motion-compensated two-stage hybrid error concealment technique is applied to the frame. The “first stage” applies Shirani’s “temporal” concealment to find a block in the previous frame with border pixels that have the smallest sum of “border errors” relative to surrounding blocks, as taught by Shirani. Ex.1005, 242-245; Ex.1006, 1-3; Ex.1003, ¶199. The resulting block is an “initial estimate” of the erroneous/missing block.

Then, the “second stage” refines the “initial estimate” by applying “spatial” concealment in which “the information around each missing block” is utilized to estimate pixels values in the interior of the estimated block. Ex.1005, 242-245; Ex.1006, 1-3; Ex.1003, ¶200. Here, the second-stage spatial concealment of Mualla-Shirani is modified to implement Saito’s PHLCT to interpolate the interior pixels of the Mualla-Shirani initial estimate X_i . Ex.1003, ¶200. The second-stage spatial concealment obtains a solution to Poisson’s equation with Neumann boundary conditions equal to the normal derivatives at boundaries of the missing block (as taught by Saito). Ex.1003, ¶200. These normal derivatives at the boundaries are based on Shirani’s “border errors” obtained in Shirani’s first stage. Ex.1003, ¶200. As a result, the Mualla-Shirani-Saito combination uses a Poisson-

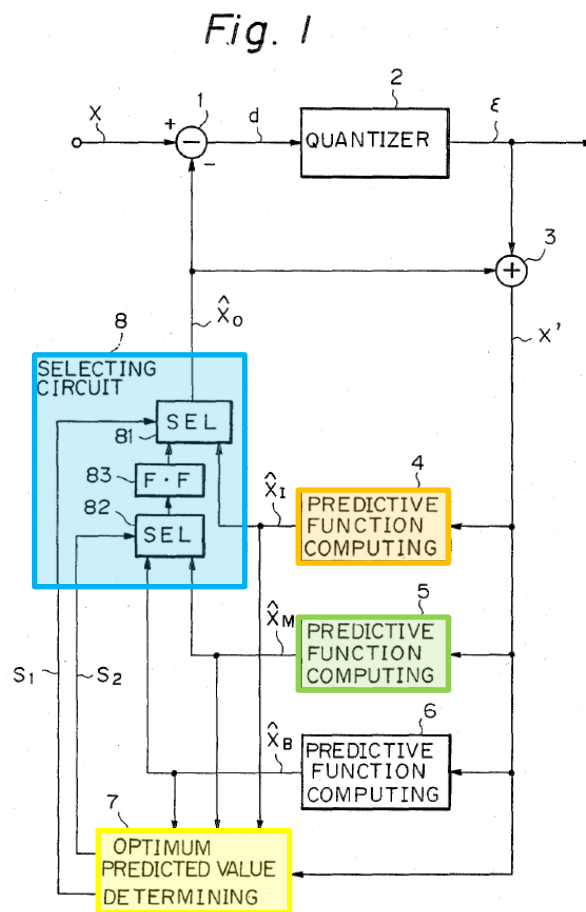
equation-solving image approximation technique (e.g., PHLCT as taught in Saito) to implement spatial error concealment in the second stage of the Mualla-Shirani two-stage hybrid error concealment technique, thereby generating a refinement of the “initial estimate” that was obtained in Shirani’s first stage. Ex.1003, ¶200. The result is an estimated frame in which erroneous/missing blocks have been concealed. Ex.1003, ¶200. As Dr. Reader explains, a POSITA would have recognized and found obvious that Saito’s image approximation techniques based on solving Poisson’s equation would have enhanced Shirani’s “second stage” techniques, “improv[ing] the video reconstruction quality” in order to achieve similar goals of image approximation. Ex.1007, 27; Ex.1006, Abstract, 3; Ex.1003, ¶200.

Furthermore, as discussed above, the modified two-stage hybrid error concealment of the Mualla-Shirani-Saito combination is implemented in both the encoder and the decoder, as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶201.

6. Takenaka (Ex.1023)

Takenaka describes an invention relating to “a differential coding apparatus adaptable for the differential coding of a fast video signal.” Ex.1023, 1:8-10. Takenaka’s “differential coding apparatus” employs “intraframe.interframe differential coding,” which was known in the art. *Id.*, 1:16-19. The apparatus includes “an intraframe prediction circuit and an interframe prediction circuit, and

adaptively selects one of these circuits in response to the movement of a picture to be coded.” *Id.*, 1:19-22. The signal that is output by the selected circuit “can be used as an optimum predicted value.” *Id.*, 1:22-24. Furthermore, “the process at the predicted value producing means,” which include the intraframe prediction circuit (annotated below in orange) and the interframe prediction circuit (annotated below in green), “and the processing at the determining means,” which includes an “optimum predicted value determining circuit” (annotated below in yellow), “are carried out in parallel.” *Id.*, 1:51-61; Ex.1003, ¶181.



Ex.1023, Fig. 1 (annotated and color added)

The predictive function computing circuits “produce predicted values X_I , X_M , and X_B ,” which are “obtained by a different prediction manner,” e.g., via intraframe or interframe coding. *Id.*, 2:30-31, 2:52-68. The optimum predicted value determining circuit determines “the optimum predicted value X_0 from predicted values X_I , X_M , and X_B . . . to output selecting signals S1 and S2.” *Id.*, 2:34-38. Then, “a selecting circuit,” which “comprises two selectors 81 and 82,” selects “the optimum predicted value from predicted values X_I , X_M , and X_B in response to selecting signals S1 and S2.” *Id.*, 2:39-43. The selecting circuit then outputs the resulting optimum predicted value “to the subtracter 1 and the adder 3.” *Id.* This selection process is designed to increase “the processing speed of the differential coding apparatus” by efficiently determining the most suitable prediction method. Ex.1023, 7:1-10; Ex.1003, ¶182.

7. Combination of Mualla-Shirani-Saito and Takenaka

As discussed above, the Mualla-Shirani-Saito combination implements a two-stage (hybrid) error concealment technique that processes images through techniques that solve Poisson’s equation. *See* § IV.A.5. The Mualla-Shirani-Saito combination also includes an “encoder” that uses “motion estimation (ME)” and “motion compensation (MC)” to generate a prediction frame, then subtracts the prediction from the current frame “to produce an error signal,” or “DFD,” which is encoded using a “transform encoder.” *Id.* The Mualla-Shirani-Saito combination

then outputs both the encoded DFD and encoded motion information. *Id.*;
Ex.1003, ¶202.

A POSITA would have found it obvious, in view of the teachings of Mualla, that the feedback-based error resilience at the encoder and decoder can be implemented such that it selects between different mechanisms for and outputs from encoding (or decoding) operations that are executed simultaneously, such as intraframe prediction or interframe prediction performed by Takenaka's "predictive function computing" circuits. Ex.1023, Fig. 1. Takenaka's "selecting circuit" can be implemented in the Mualla-Shirani-Saito combination to select between different encoding or decoding modes in response to errors that are detected through the two-stage (hybrid) error concealment technique. As shown below, Mualla provides a "selector" to control interframe or intraframe coding. Ex.1005, 72, Fig. 3.1. As Dr. Reader explains, using a selector to switch between normal encoding or decoding and error concealment "would have been a natural extension of the techniques disclosed within Mualla." Ex.1003, ¶203. And Takenaka's teachings, which further describe such a selector, are within the general knowledge of a POSITA; as Dr. Reader explains, dual-mode prediction and selecting between two predictions were known as early as 1958. Ex.1003, ¶203 (citing, e.g., Graham (Ex.1027), Fig. 6).

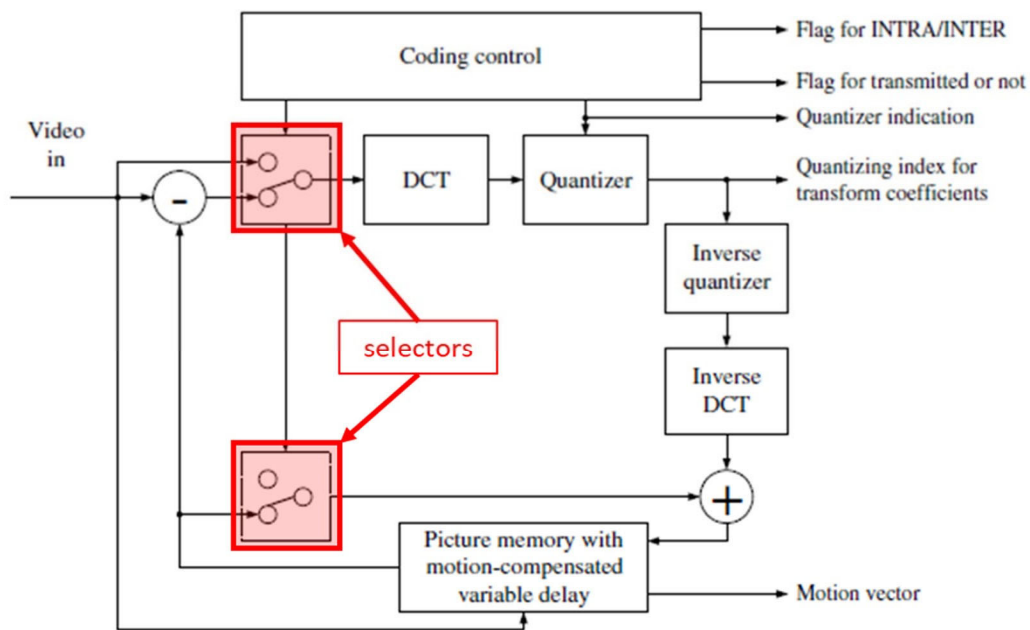


Figure 3.1: H.263 video encoder

Ex.1005, FIG. 3.1 (annotated and color added).

Although Takenaka’s “selecting circuit” is primarily described with respect to an encoder (a “coding apparatus,” Ex.1023, 2:19-21), a POSITA would have recognized that its teachings of a “selecting circuit” are applicable to decoding operations (as recited in certain challenged claims), as encoding and decoding are complementary processes that are expected to work similarly. Indeed, a POSITA would have recognized that selecting circuits in both encoders and decoders were known prior to the Critical Date. Ex.1003, ¶204 (citing, e.g., Ex.1026, Figs. 2, 8).

A POSITA would have found it obvious or understood that Mualla’s “selector” could be modified with Takenaka’s “selecting circuit” to select between predictions, inter and intra coding, as well as between normal coding and error

concealment. Further, Takenaka's selecting between intra and inter coding prediction outputs generated in parallel itself minimizes prediction error by selecting the optimum method after comparing the prediction errors to select the value with the least error. Ex.1023, 1:16-25, 2:19-43. As Dr. Reader explains, a POSITA would have understood that Takenaka's selecting mechanism is particularly effective in addressing the goals emphasized by Mualla (and Shirani and Saito) for error concealment applications, and further, that a POSITA would have recognized a reasonable expectation of success in incorporating Takenaka's teachings. Ex.1003, ¶205.

B. Claims 6-8, 10 (“Moving picture decoding”)

As an initial matter, the '025 patent includes claims that recite both encoding and decoding operations (despite individual claims reciting an “encoding” or “decoding” apparatus in the preamble). *See, e.g.*, Ex.1001, Claim 6. To indicate which limitations include predominately encoding or decoding steps, Petitioner has added the annotations of ENC (“encoding”) and DEC (“decoding”) to the title of each limitation. This annotation is solely an aide to the reader and should not be viewed as an argument that a limitation discloses only one operation or the other. Ex.1003, ¶¶207-208.

1. [6.pre]

Mualla teaches extensive details of an H.263-based video codec system, which implements a *decoding apparatus* for decoding *moving pictures* (e.g., “video”). Ex.1005, 23-26, 31, 36, 51-52, 227-228, 241-242, 265. For example, in describing “Transform Coding” which “forms the basis of almost all video coding standards,” Mualla describes a *decoder* implemented in a “transform coding system.” Ex.1005, 51 (last paragraph), Figure 2.9 (reproduced below). As another example, in describing techniques for “error resilience” in video coding systems, Mualla discloses a *decoder* implemented in a “typical video communication system.” Ex.1005, 227-228, Figure 9.1; Ex.1003, ¶209.

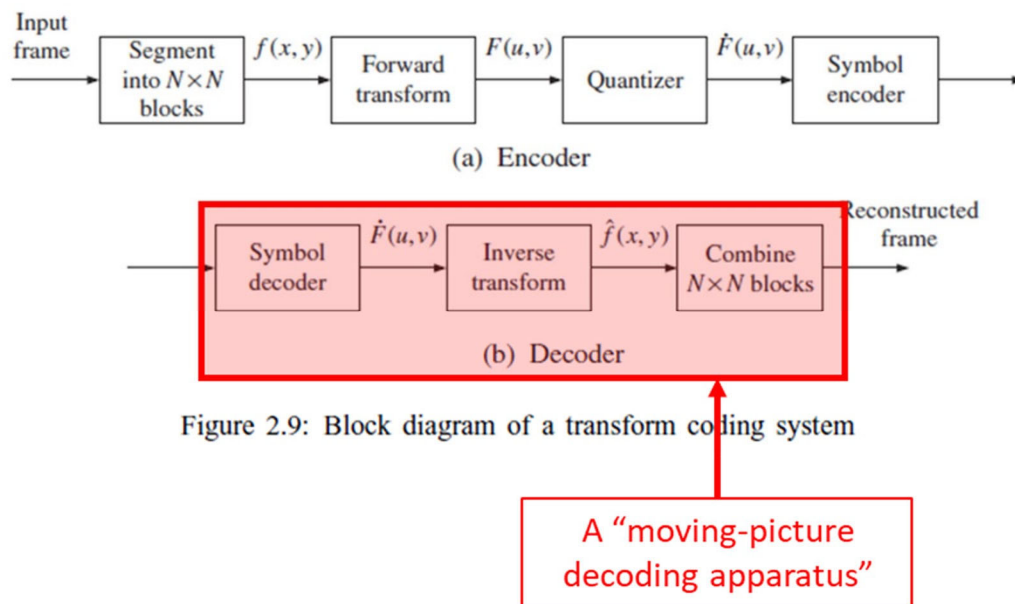


Figure 2.9: Block diagram of a transform coding system

Ex.1005, FIG. 2.9 (annotated and color added)

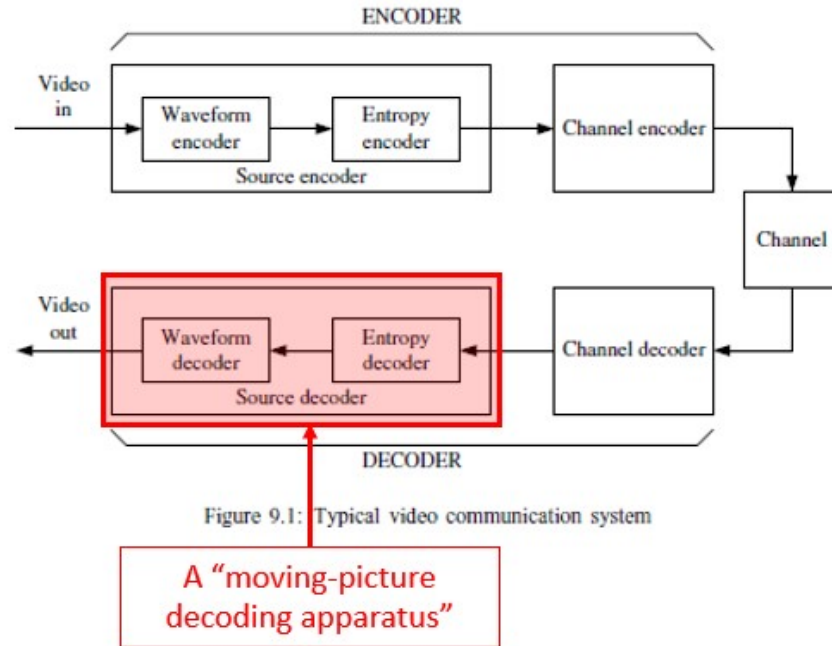


Figure 9.1: Typical video communication system

Ex.1005, FIG. 9.1 (annotated and color added)

Shirani similarly discloses using the H.263-based video codec system and its associated *decoding apparatus*. Ex.1006, 2-4 (“[w]hen channel errors occur, **the decoder** usually discards the erroneous data”), 4 (“H.263”); Ex.1003, ¶210.

2. [6.1.DEC]

Shirani discloses that “[c]oded video information [(“*coded data from an input signal*”)] is first grouped into packets” and the “video packets are ... multiplexed with audio information” using an “H.223 multiplexing” protocol. Ex.1006, 5. As Dr. Reader explains, multiplexing video packets at an encoder would have required a means for “*demultiplex[ing]*” the video packets at the decoder, for example, by using a *demultiplexer*. Ex.1003, ¶211. Indeed, the H.263 standard (the standard used by Mualla and Shirani) corroborates the use of

demultiplexers in H.263 codecs for “*demultiplex[ing] coded data from an input signal.*” Ex.1016, 2.

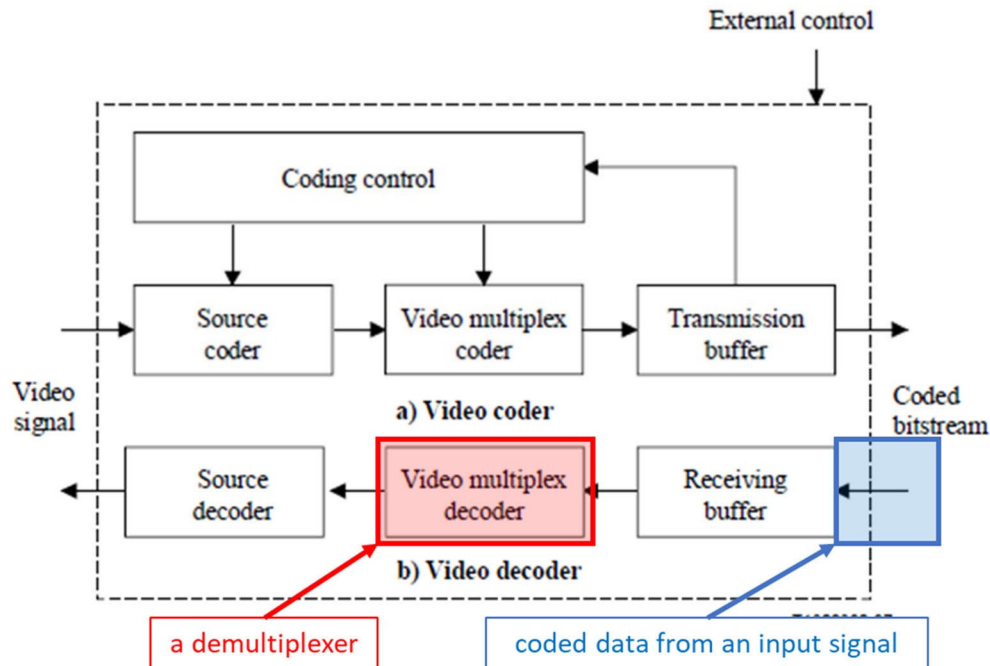
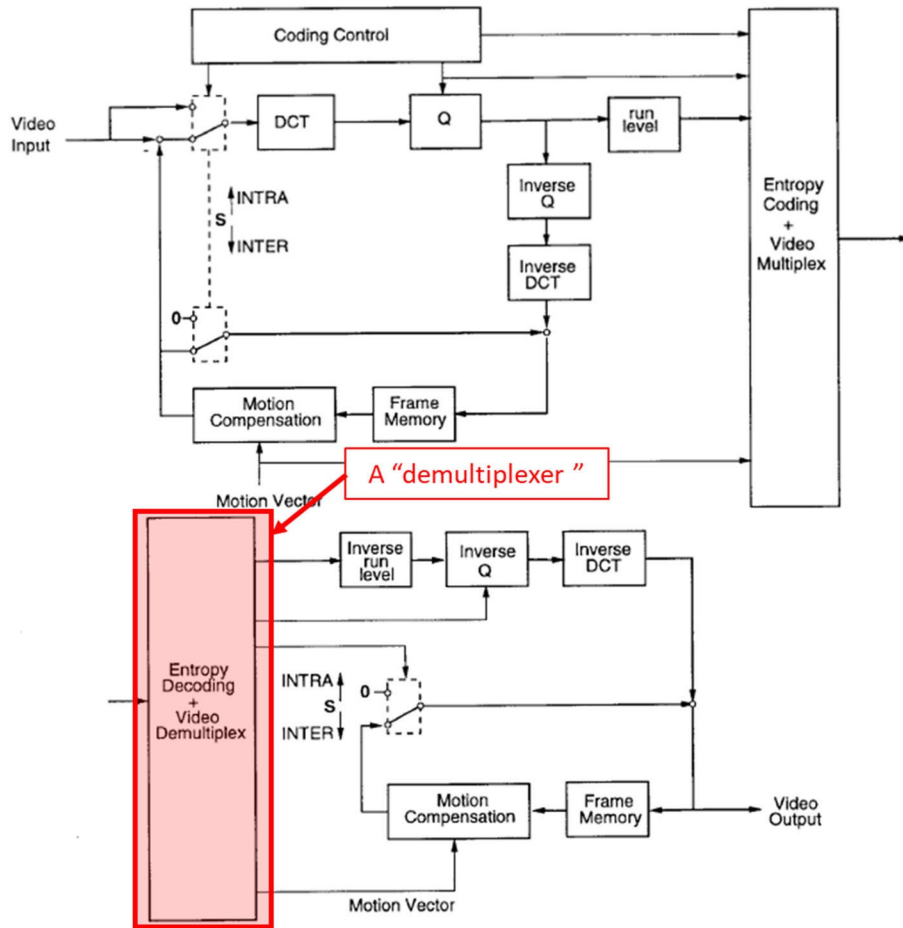


Figure 1/H.263 – Outline block diagram of the video codec

Ex.1016, FIG. 1 (annotated and color added).

Additionally, Girod – a H.263 codec reference cited by Mualla as reference 201 – discloses that information is “multiplexed to the video bitstream.” Ex.1008, 2, FIG. 1(b); Ex.1003, ¶212. FIG. 1(b) illustrates an example decoder with a “*demultiplexer.*”



Ex.1008, FIG. 1(a) (“encoder”) and 1(b) (“decoder”) (annotated and color added).

Furthermore, Mualla and Shirani both disclose the H.263 *specific syntax structure*. Ex.1005, 93, 232; Ex.1006, 2, 4-5. Indeed, Mualla refers to the H.263 codec standard as “the *syntax* of H.263.” Ex.1005, 93. Mualla and Shirani also both use the specific syntax structure of the H.263 codec standard as a gauge for detecting errors in a video bitstream. Ex.1005, 231-232; Ex.1006, 2. For example, Mualla’s process of “error correction” uses “*syntax* ... violations” to detect errors in a video bitstream. Ex.1005, 231-232. Shirani discloses a similar process where

a decoder determines the “location of missing blocks” by “detecting ... semantic or *syntactic* violations.” Ex.1006, 2; Ex.1003, ¶213.

3. [6.2.ENC]

As described above in 6.1.DEC, Shirani discloses *multiplexing a coded bitstream*, and a POSITA would have understood or found obvious that the H.263 codec standard, as used in Mualla and Shirani, included *multiplexing* operations. Ex.1006, 5; Ex.1003, ¶214; *see supra*, [6.1.DEC]. Indeed, the H.263 standard discloses multiplexing and demultiplexing a “*coded bitstream*.” Ex.1016, 2, 9; Ex.1003, ¶214.

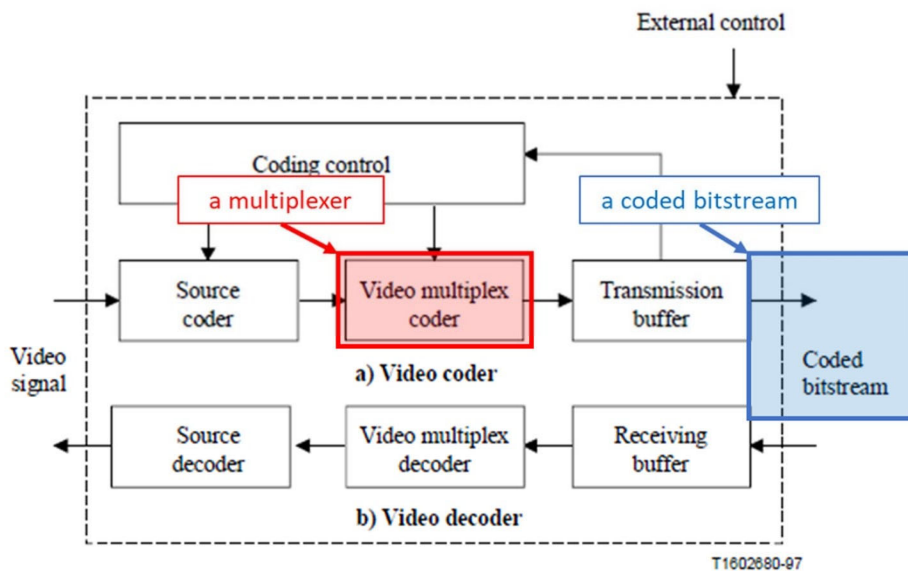


Figure 1/H.263 – Outline block diagram of the video codec

Ex.1016, FIG. 1 (annotated and color added).

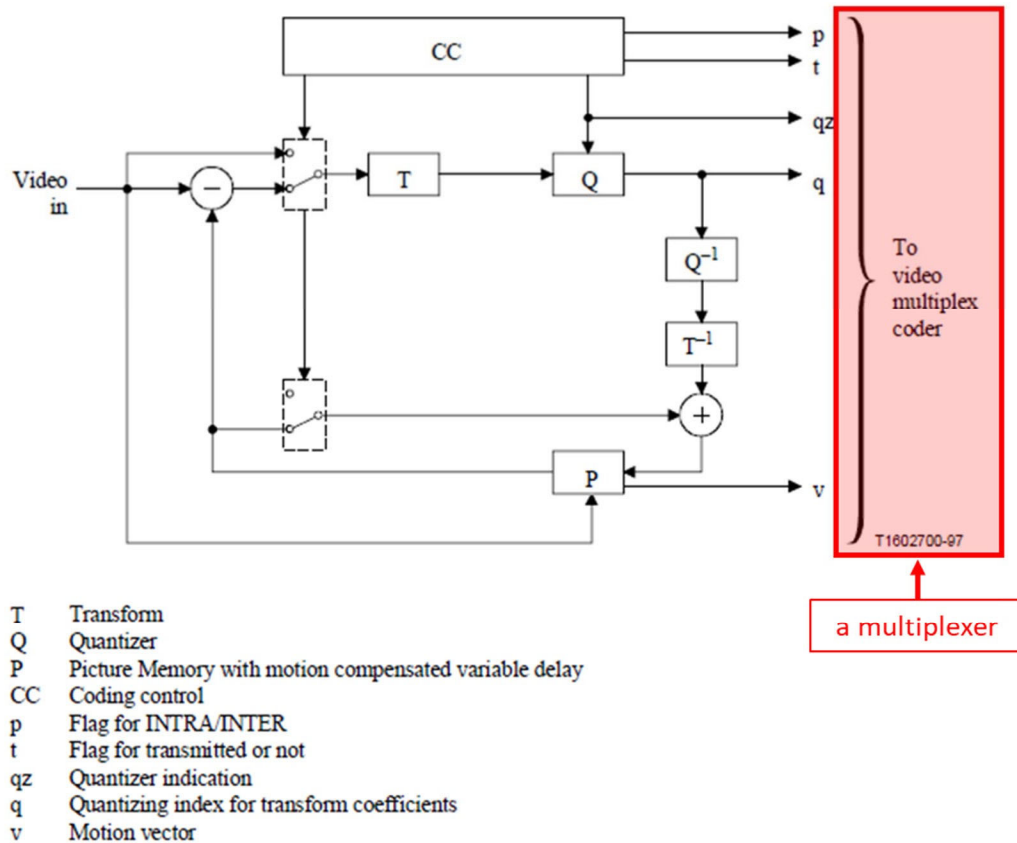


Figure 3/H.263 – Source coder

Ex.1016, FIG. 3 (annotated and color added).

Furthermore, as shown in Mualla’s Figure 3.1 and explained in further detail below, a POSITA would have understood or found obvious, based on the teachings of Mualla, that the coded bitstream is obtained by encoding motion vectors (which include the claimed “*border motion-vector data*”) and a quantization index (the claimed “*post-quantization data*”), along with various coding control information. Ex.1005, Figure 3.1, 71-75; Ex.1003, ¶215.

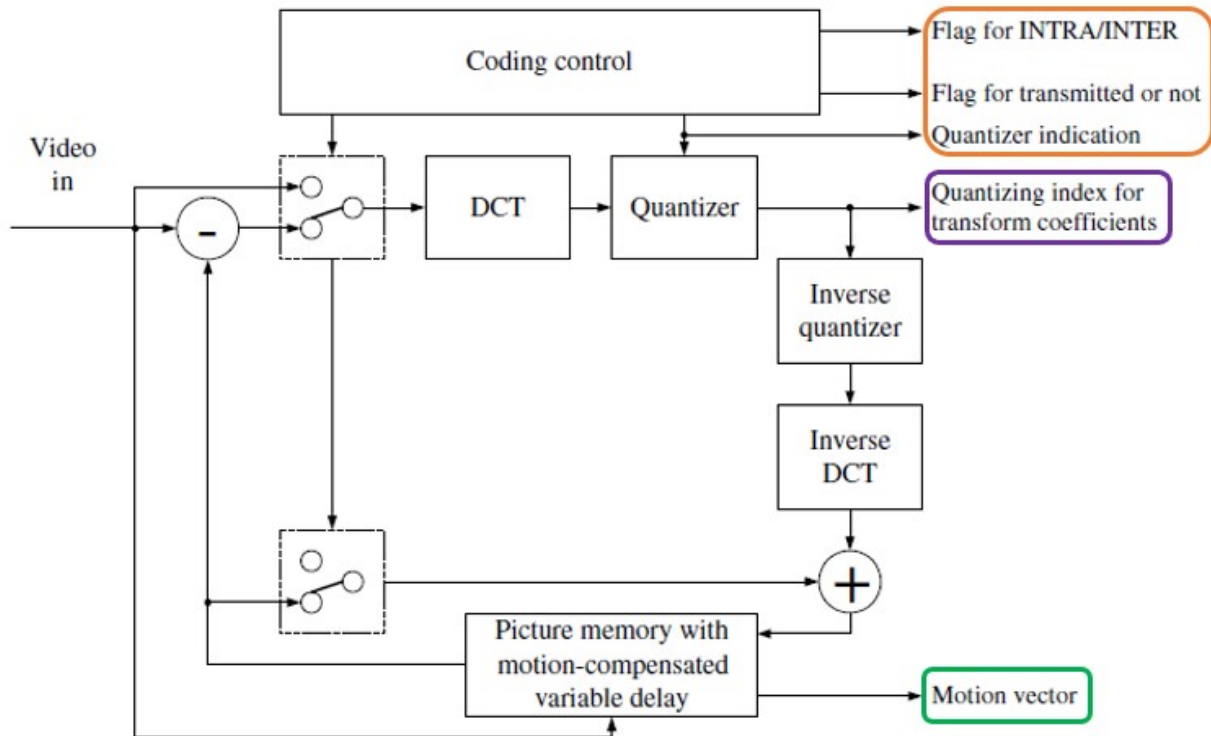


Figure 3.1: H.263 video encoder

...obtained by predictive coding

Mualla describes numerous operations at the encoder that perform predictive coding. Ex.1003, ¶216. For example, Mualla discloses a “video codec” that includes a “motion estimation (ME) block ... to estimate the movement that occurred between the current frame and a reference frame.” Ex.1005, 26-27, 115-117, FIG. 1.1. Mualla refers to this estimation as “motion information.” *Id.* Motion information is then “utilized by the motion compensation (MC) block to move the contents of the reference frame to provide a prediction of the current frame” (“*predictive coding*”) which Mualla also refers to as the “motion-compensated prediction (MCP).” *Id.* Mualla refers to “motion compensated

prediction” as a type of “predictive coding” in which regions of an input frame are predictively coded “with respect to the corresponding regions in the reference frame.” Ex.1005, 61; Ex.1003, ¶216. Mualla’s “typical video codec” illustrates these components.

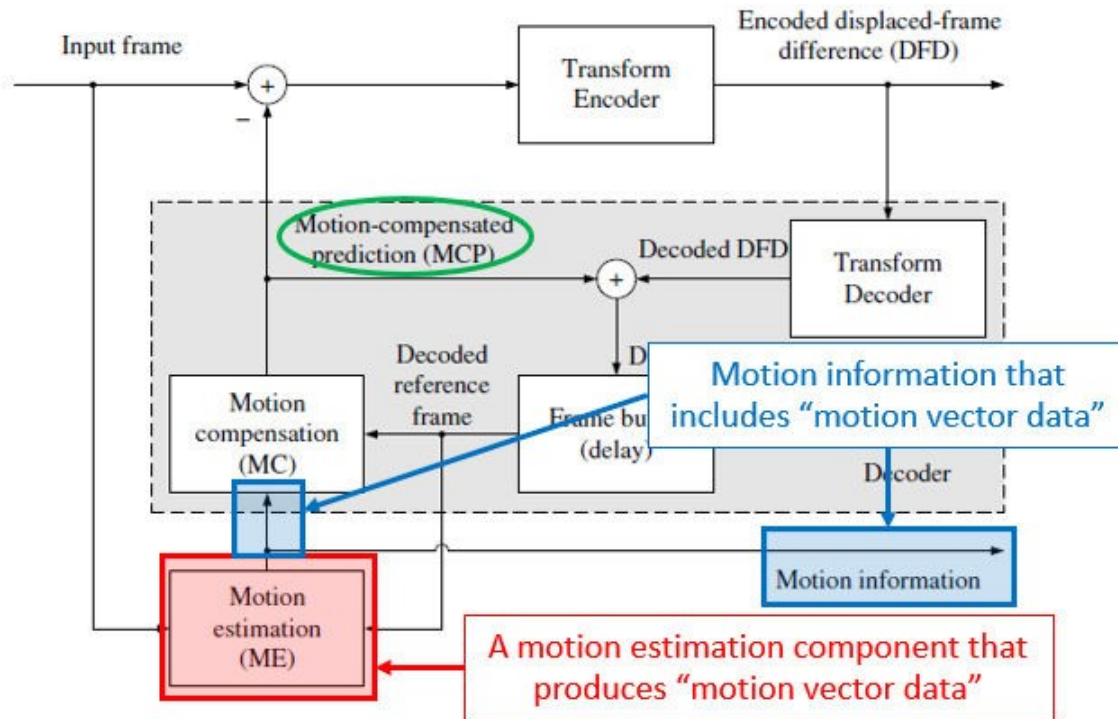


Figure 1.1: Typical video codec

Ex.1005, FIG. 1.1 (annotated and color added).

As another example of predictive coding, Mualla discloses that the “motion information” (motion vector) itself is predictively coded. Mualla discloses that the “motion of the [a] current block” (e.g., the motion information) is “represented by a motion vector” (“*motion-vector data*”) which is the “displacement between the block and its best-match block in the reference frame.” Ex.1005, 62, FIG. 2.16; Ex.1003, ¶217.

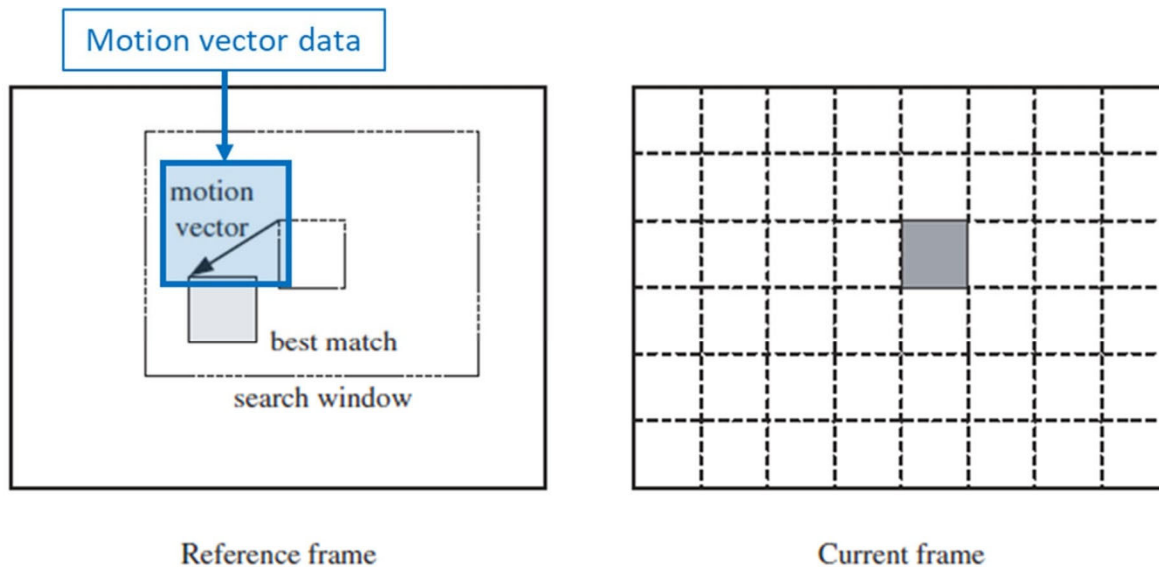


Figure 2.16: Block-matching motion estimation

Ex.1005, FIG. 2.16 (annotated and color added).

Mualla discloses that the “estimated motion vector” is “predictively coded” (part of “*a coded bitstream obtained by predictive coding*”), meaning that the “motion vector difference $MVD = MV - MVP$ is encoded instead of the motion vector itself.” Ex.1005, 74, *see also* 115-117; Ex.1003, ¶218.

Border motion-vector data

As an initial matter, the '025 patent describes “***border motion-vector data***” as “data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found.” Ex.1001 2:59-61; *see infra*, [6.6.ENC]; Ex.1003, ¶219.

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in the

encoder (as well as the decoder), as taught by Mualla. Ex.1005, 246-250. The encoder implements the first stage of Shirani's two-stage error concealment technique to the motion of a missing/erroneous macroblock by performing temporal concealment. *See supra*, §IV.A.2. In the combination, Shirani's first stage uses motion vectors of neighboring blocks to obtain estimates of pixels at the borders of a missing block. *See supra*, §IV.A.2; Ex.1006, 2-3. In particular, the first stage of Shirani estimates motion vectors of subblocks adjacent to the missing block, and in doing so, utilizes "candidate motion vectors" ("data on a motion vector") which can include motion vectors of "neighboring blocks." *See supra*, §IV.A.2; Ex.1006, 2-3; Ex.1003, ¶220. These motion vectors of neighboring blocks are used to obtain an estimate of pixels at each border of the missing block ("from a border of the rectangular zone in the picture to be coded"), by minimizing the "squared sums of border errors" (in total, "***border motion-vector data***" or, "data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found"). *Id.*

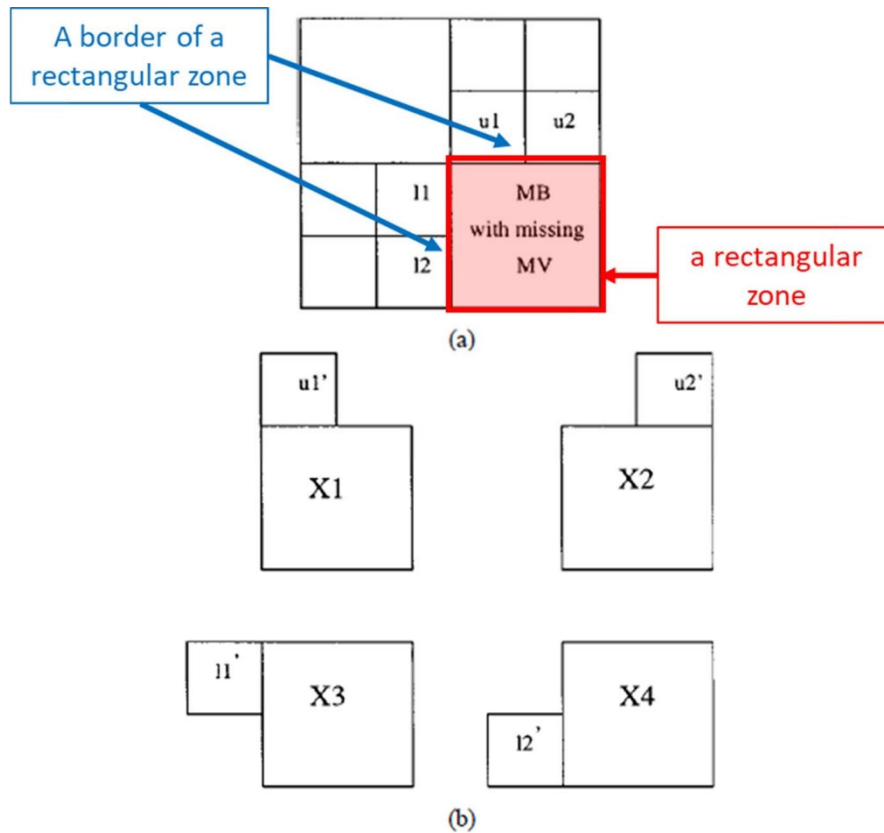


Fig. 1. (a) The four subblocks, (b) their corresponding subblocks in the previous frame, and the blocks connected to them.

Ex.1006, FIG. 1(a) (annotated and color added).

Thus, Shirani's 1st stage obtains "motion vectors" representing a spatial "displacement" (data representing the difference in spatial position) between (i) smaller-sized subblocks $u1$, $u2$, $l1$, $l2$ adjacent to each border of the missing block (a border of the block to be predicted in the picture to be coded) and (ii) corresponding smaller-sized subblocks $u1'$, $u2'$, $l1'$, $l2'$ in the previous frame that minimize the border errors (and the best-matched border in the reference picture) and discloses "border motion-vector data" as this term was previously interpreted by Patent Owner. Ex.1006, 3, Fig. 1(a). Shirani defines "border error" in "terms

of pixels,” including “bottom line pixels of the block to the top of the missing block” and “right column pixels of the block to the left of the missing block” and pixels adjacent to “the subblocks ... to the right and below of the missing block.”

Id.; see also Ex.1003, ¶¶177, 221, 229.

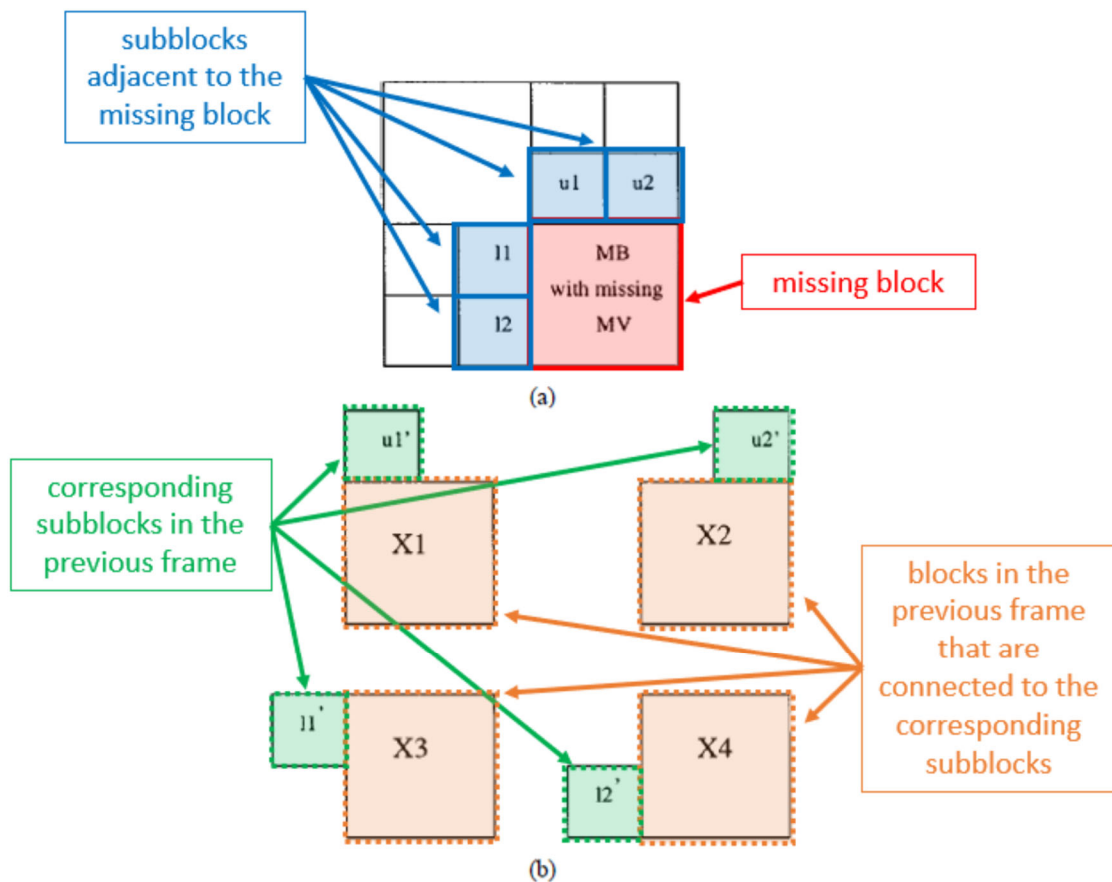


Fig. 1. (a) The four subblocks, (b) their corresponding subblocks in the previous frame, and the blocks connected to them.

Ex.1006, FIG. 1(a) (annotated and color added)

After Shirani’s **first stage** estimates pixels at the **borders** of the missing block, the **second stage** of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-

Takenaka) combination estimates pixels in the **interior** of the missing block using Saito's technique for solving Poisson's equation with the border pixels (specifically, Shirani's "border errors" of the missing block) providing boundary conditions in Saito's techniques. *See supra*, §IV.A.2; *see infra* [6.7.ENC]; Ex.1003, ¶221. Therefore, because Shirani's first stage temporal concealment uses the motion vectors of neighboring blocks to estimate only the motion of pixels at the border of the missing block, Shirani discloses the claimed "***border motion vector data***."

Post-quantization data obtained by quantization in the predictive coding

Mualla discloses that "quantization is a key element of a video coding system" and illustrates a "quantizer" in its example "video encoder" and "transform coding system." Ex.1005, 40-42, 52, FIGS. 2.3, 2.9, 3.1. Mualla discloses that, in "transform coding" (also known as "DCT coding"), "coefficients can be quantized according to their visual importance" ("***post-quantization data***"). Ex.1005, 52, 75; Ex.1003, ¶222.

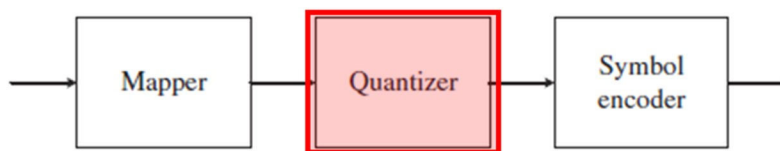


Figure 2.3: Elements of a video encoder

Ex.1005, FIG. 2.3 (annotated and color added).

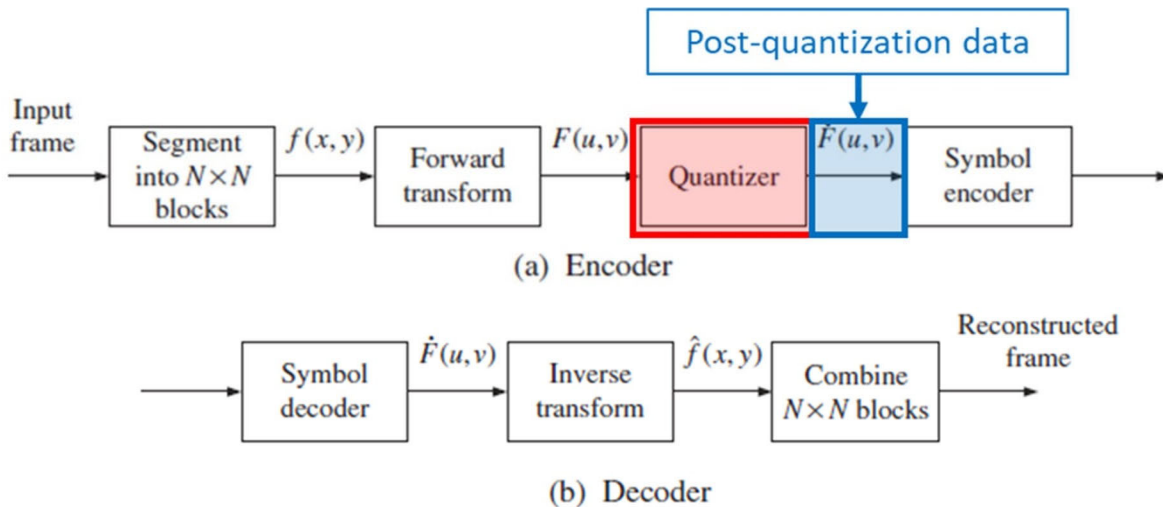


Figure 2.9: Block diagram of a transform coding system

Ex.1005, FIG. 2.9 (annotated and color added).

Mualla discloses that these coefficients are derived by obtaining a “displaced frame difference (DFD)” that represents “the coded prediction error signal” (“*the predictive coding*”). Ex.1005, 61, FIG. 2.15. The DFD signal can be coded “using any of the intraframe coding methods discussed in Section 2.6” and “the most commonly used method is transform coding, in particular block-based DCT transform coding.” Ex.1005, 62. Mualla discloses that the DCT transform “is applied to the DFD values” and that this “results in ... transform coefficients for each [macro block].” Ex.1005, 74. These coefficients are then “quantized using a uniform scalar quantizer” (“*post-quantization data obtained by quantization in the predictive coding*”). Ex.1005, 75, FIG. 3.1; Ex.1003, ¶223. This process is illustrated in Mualla’s example “H.263 video encoder.”

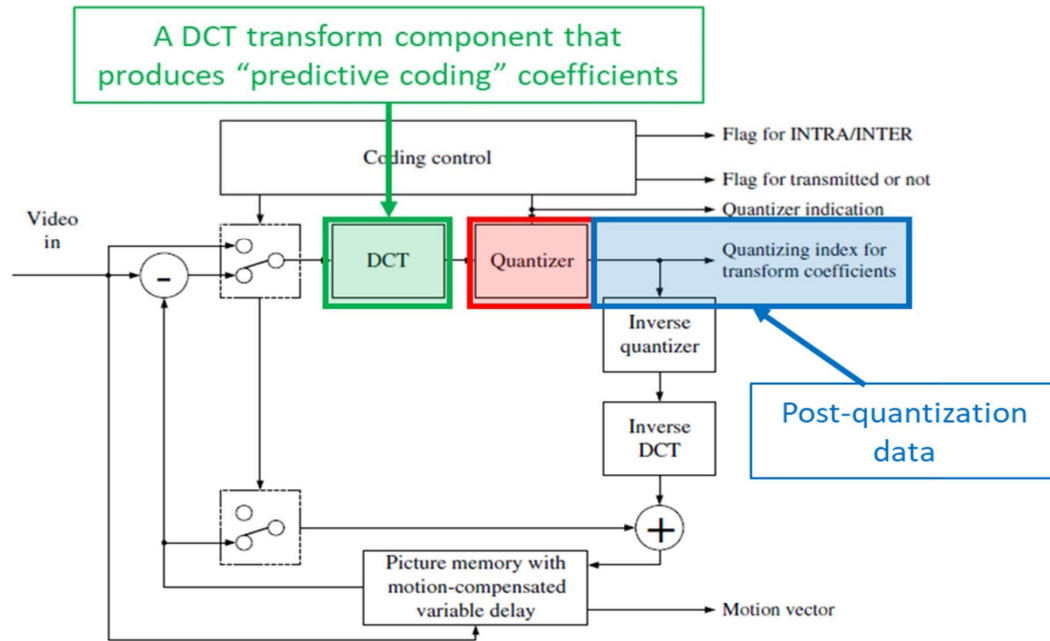


Figure 3.1: H.263 video encoder

Ex.1005, FIG. 3.1 (annotated and color added).

4. [6.3.ENC]

As discussed above in [6.2.ENC], Mualla's DFD represents "the coded prediction error signal" ("*residual picture that is a residual signal*") which is created by subtracting a "motion compensation prediction (MCP)," produced from the "reference frame" ("*a predictive picture produced from a reference picture that is a local decoded video signal*"), from the "current frame" ("*a picture to be coded that is an input moving-picture video signal to be subjected to coding*").

Ex.1005, 26-27, FIG. 1.1; Ex.1003, ¶224. Mualla's "typical video codec" illustrates this process.

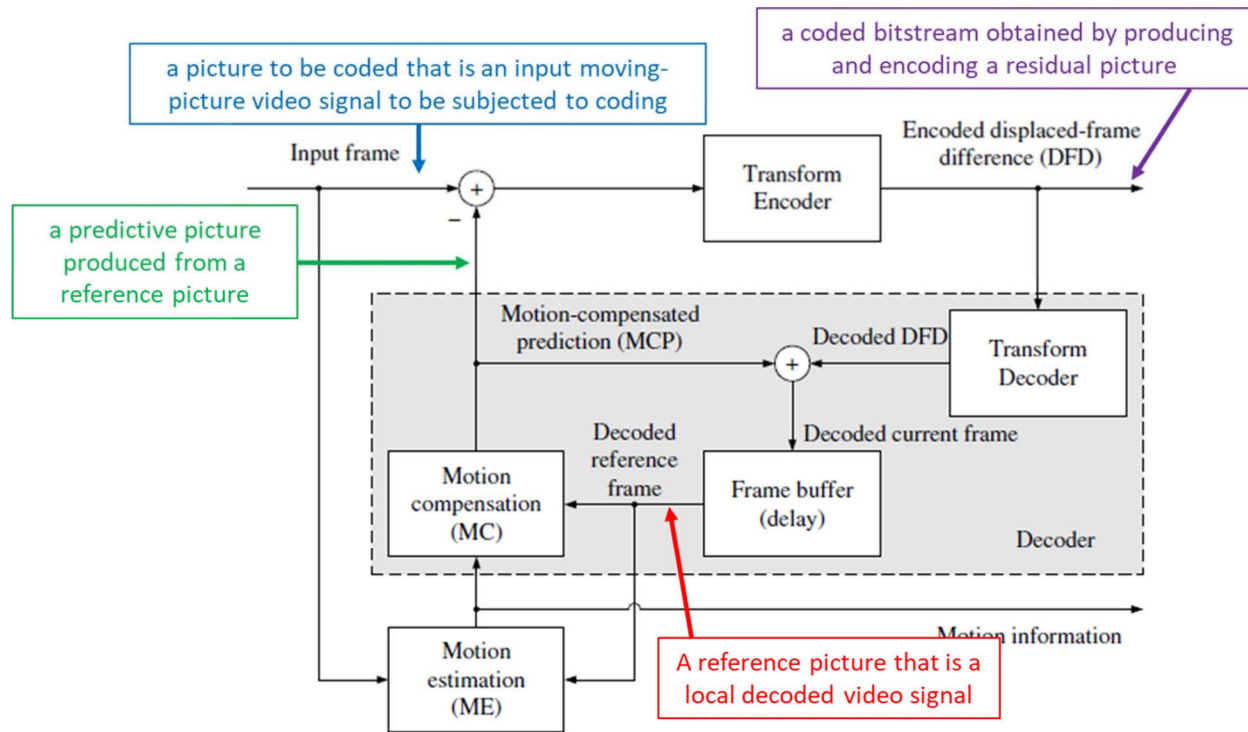


Figure 1.1: Typical video codec

Ex.1005, FIG. 1.1 (annotated).

Mualla also discloses that an input video signal “consists of a sequence of pictures (or frames)” where “[e]ach picture is divided into groups of blocks (GOBs)” and each GOB “is divided into macroblocks (MBs)” (“*each of a plurality of rectangular zones*”). Ex.1005, 71-72. These MBs are further divided into individual “pixels,” or “pels” (“*each composed of a specific number of pixels, into which a video area of the moving-picture video signal is divided*”). Ex.1005, 36, 71-72, FIG. 3.2; Ex.1003, ¶225. FIG. 3.2 of Mualla illustrates this breakdown.

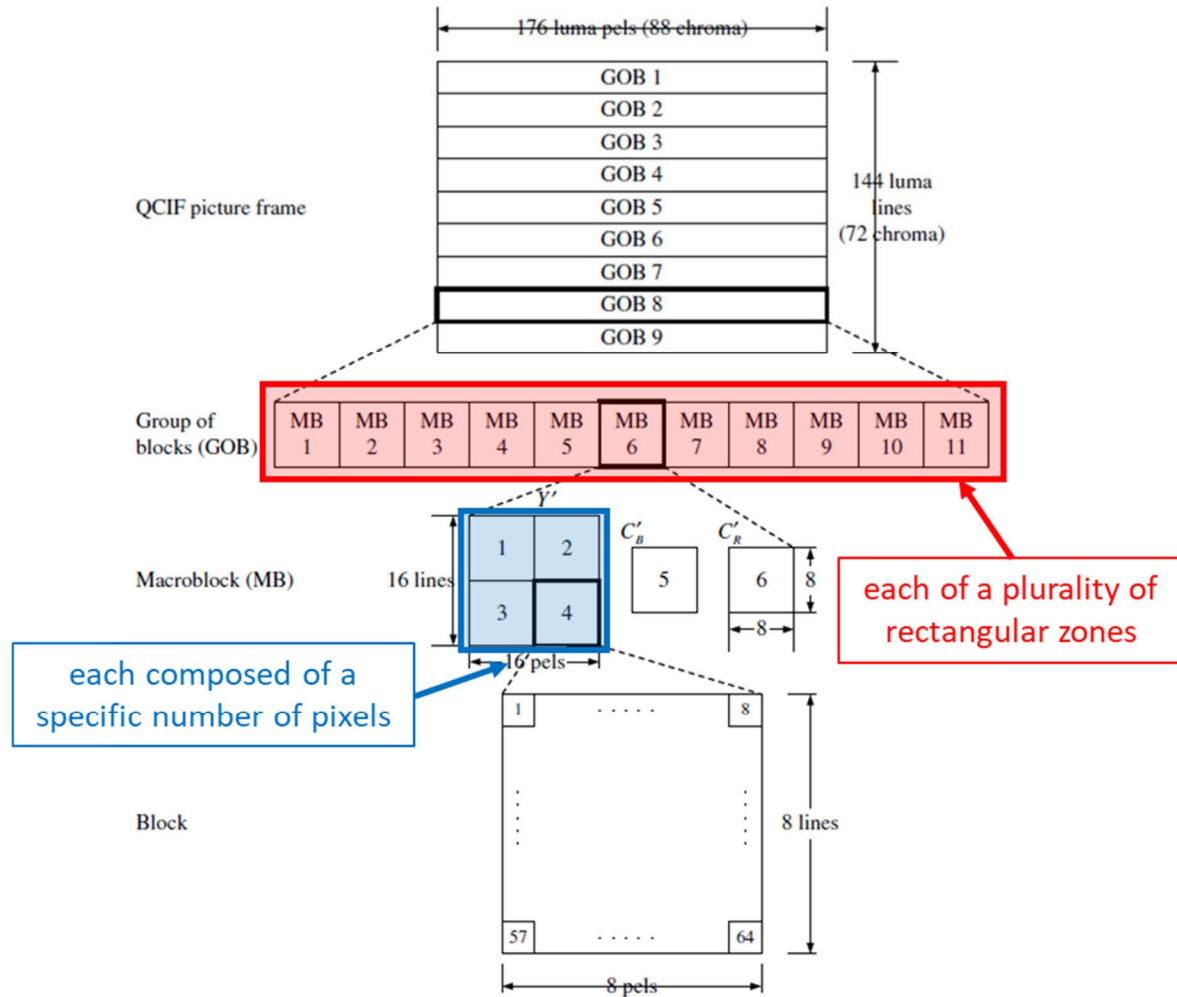


Figure 3.2: H.263 picture structure for a QCIF frame

Ex.1005, FIG. 3.2 (annotated and color added).

5. [6.4.ENC]

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in the **encoder** (as well as the decoder) using “interactive techniques,” as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶226.

For the first stage, as discussed above in [6.2.ENC], Mualla discloses “boundary matching (BM)” to select a “suitable replacement [motion vector] from a set of candidate [neighboring] motion vectors. Ex.1005, 245; *see supra* [6.2.ENC]. Shirani’s first stage applies a similar technique to estimate a missing motion vector by considering “subblocks adjacent to the missing block.” Ex.1006, 2-3, FIG. 1(a); *see supra* [6.2.ENC]; Ex.1003, ¶227.

In more detail, Shirani explains that in order to obtain an “initial estimate” for each of the missing blocks (“*the rectangular zones*”), “subblocks adjacent to the missing block, i.e., *u1*, *u2*, *l1*, and *l2* in Fig. 1(a), are considered” (“*another plurality of rectangular zones adjacent to the rectangular zones*”). Ex.1006, 3, FIG. 1(a) (reproduced below); Ex.1003, ¶228.

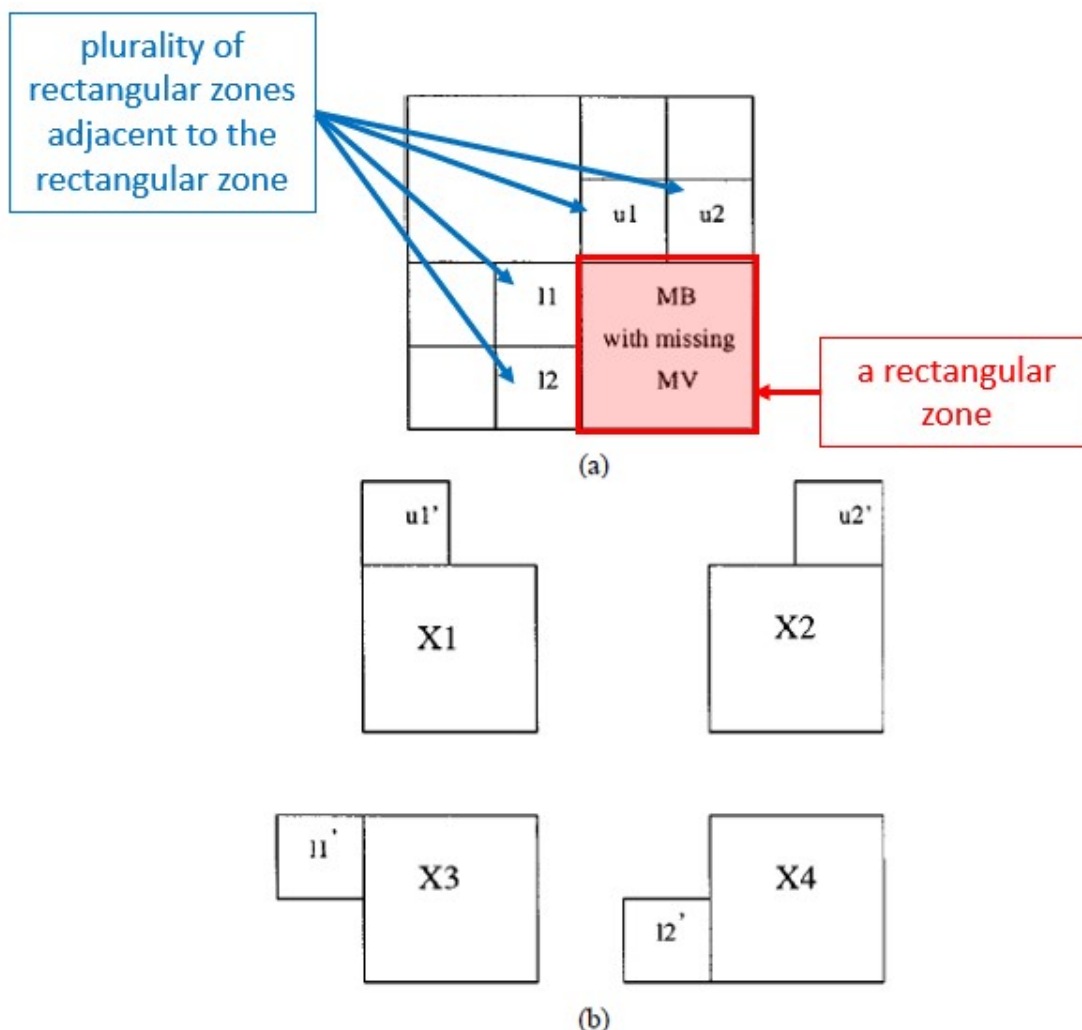


Fig. 1. (a) The four subblocks, (b) their corresponding subblocks in the previous frame, and the blocks connected to them.

Ex.1006, FIG. 1(a) (annotated and color added).

As explained in Section IV.A.2, above, Shirani discloses that “for each of these subblocks [$u1$, $u2$, $l1$, and $l2$]], a corresponding subblock in the previous frame is determined” (denoted $u1'$, $u2'$, $l1'$, and $l2'$ at the bottom of FIG. 1(a), above). Ex.1006, 3. Then, “four blocks, namely $X1$, $X2$, $X3$, and $X4$, which are connected to $u1'$, $u2'$, $l1'$, and $l2'$ (respectively), are determined” and “a block from the above four blocks [$X1$, $X2$, $X3$, and $X4$]] that minimizes the squared

sums of border errors, between the estimated block and its adjacent above and left blocks, is selected.” Ex.1006, 3; Ex.1003, ¶229. Shirani achieves this by defining “border error[s]” in “terms of pixels,” including an error for the “bottom line pixels of the block to the top of the missing block” and an error for the “right column pixels of the block to the left of the missing block” and applying the same technique to define border errors for pixels adjacent to “the subblocks ... to the right and below of the missing block” (“*obtaining a boundary condition of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones*”). *Id.* Furthermore, although the above description focused on adjacent subblocks above and to the left of the missing block, Shirani also discloses that “the same method can be extended to include the subblocks of the blocks to the right and below of the missing block” (“... *of each of a plurality of borders between the rectangular zones and another plurality of rectangular zones adjacent to the rectangular zones*”). *Id.*

FIG. 5 of Shirani illustrates an example of concealing errors for multiple damaged macroblocks (a “*plurality*” of “*rectangular zones*”) in a frame. Ex.1006, FIG. 5; Ex.1003, ¶230.

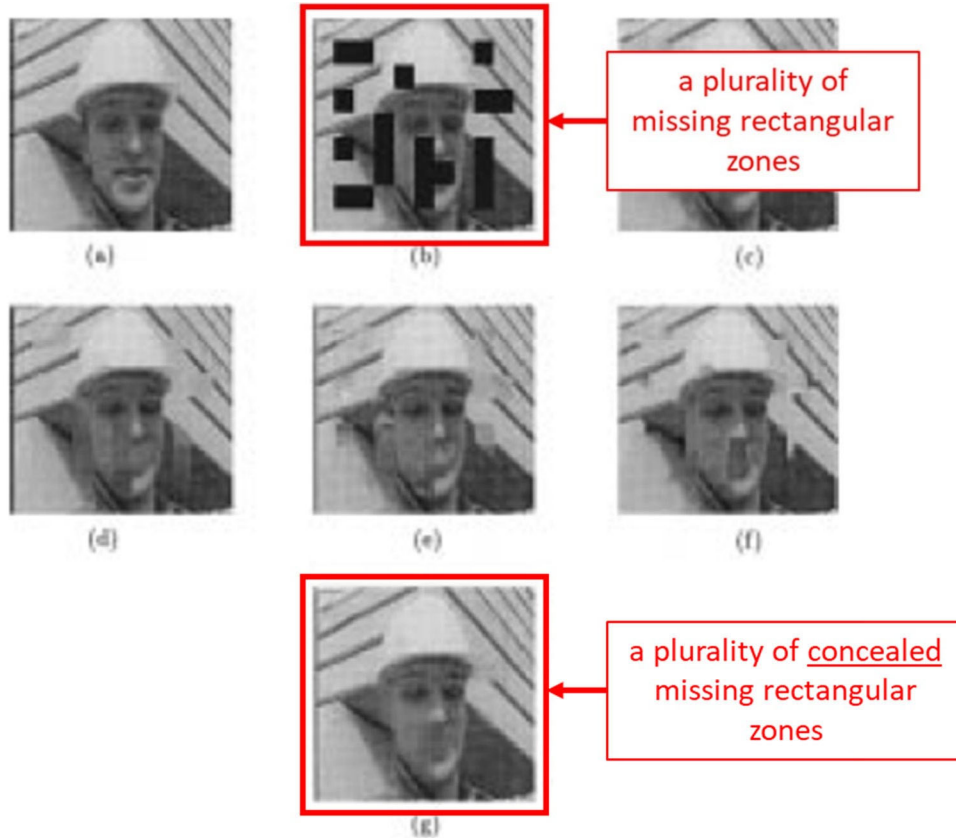


Fig. 5. A frame from the video sequence Foreman: (a) original; (b) missing blocks; reconstructed using (c) a nonadaptive GMRF model; (d) a suboptimal MRF model; (e) the method proposed in [4]; (f) the method proposed in [3]; and (g) our adaptive MRF model.

Ex.1006, FIG. 5 (annotated).

6. [6.5.ENC]

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in the **encoder** (as well as the decoder) using “interactive techniques,” as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶231.

As described above in [6.4.ENC], Shirani finds and evaluates the “border error” of a plurality of neighboring subblocks when considering the suitability of

these subblocks for error concealment (“***finding a border, of the reference picture, having a boundary condition that matches the boundary condition***”). Ex.1006, 2-3, FIG. 1; *see supra* [6.2.ENC], [6.4.ENC]. Additionally, Shirani discloses that the “search for displacement of each of the subblocks is restricted to a set of candidate motion vectors,” specifically, “the motion vectors of available neighboring blocks” (“***finding a border ... by motion-vector search in the reference picture***”). Ex.1006, 3, FIG. 1(a); *see supra* [6.2.ENC], [6.4.ENC]; Ex.1003, ¶232.

7. [6.6.ENC]

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in the **encoder** (as well as the decoder) using “interactive techniques,” as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶233.

As described in [6.2.ENC], [6.4.ENC], and [6.5.ENC] above, Shirani’s first stage searches for candidate replacement motion vectors and generates border errors that define the similarity of the neighboring subblock’s motion vector (“***border motion-vector data***”). *See supra*, [6.2.ENC], [6.4.ENC], [6.5.ENC]; Ex.1003, ¶234. Indeed, Shirani’s “candidate motion vectors” are “***data on a motion vector from a border of the rectangular zone in the picture to be coded to the border of the reference picture thus found,***” at least because these candidate

motion vectors conceal missing motion vectors between “*the reference picture*” and “*the picture to be coded.*” *Id.*

Moreover, as described above, Shirani generates border errors that quantify the similarity of the neighboring subblocks’ motion vectors compared to the missing block to be concealed (“*defining a boundary condition of a border that corresponds to the border motion vector data, from the reference picture*”). *See supra*, [6.4.ENC], [6.5.ENC]; Ex.1003, ¶235. Shirani’s border errors evaluate the similarity of the neighboring subblocks’ motion vectors, so they are therefore “*based on the border motion vector data*” of the neighboring subblocks. *Id.*

8. [6.7.ENC]

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in the **encoder** (as well as the decoder) using “interactive techniques,” as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶236.

In particular, as described above in [6.6.ENC], Shirani’s first stage generates *border motion-vector data* for each damaged or missing subblock. *See supra*, [6.2.ENC], [6.4.ENC]-[6.6.ENC]. The second stage of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination uses “the information around each missing block ... to refine the initial estimate” (as taught by Shirani) by using

Saito's technique for solving Poisson's equation with known boundary conditions.

Ex.1006, 3; Ex.1007, 27; *see supra* §§IV.A.2, IV.A.3, IV.A.4-5; Ex.1003, ¶237.

As Dr. Reader explains, and as discussed above in the combination, a POSITA would have recognized and found obvious that Saito's image approximation techniques based on solving Poisson's equation would have enhanced Shirani's "second stage" techniques, "improv[ing] the video reconstruction quality" in order to achieve similar goals of image approximation. Ex.1007, 27; Ex.1006, Abstract, 3; Ex.1003, ¶238. Saito's Polyharmonic Local Cosine Transform (PHLCT) is an enhancement of a Polyharmonic Local Sine Transform (PHLST) technique for image approximation. Ex.1006, Abstract, 27. Saito discloses that the PHLST method "segments an image into local pieces using ... characteristic functions, then decomposes each piece into two components: the polyharmonic component and the residual." Ex.1006, Abstract, 4. Saito explains that the polyharmonic component is "obtained by solving the elliptic boundary value problem associated with the so-called polyharmonic equation (e.g., Laplace's equation, biharmonic equation, etc.) given the boundary values (the pixel values along the boundary created by the characteristic function)." *Id.* The PHLCT variation of PHLST approximates an image between two boundaries by "solv[ing] the ... Poisson equation with the Neumann boundary condition" ("**generating an**

estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson's Equation"). Ex.1007, 27; Ex.1003, ¶238.

In implementing Saito's teachings into the Mualla-Shirani combination, a POSITA would have recognized and found obvious that the "border errors" of Shirani are suitable to use as the "Neumann boundary condition" for Saito's PHLCT method, and that PHLCT would have been capable of approximating the image between these boundary conditions ("*thus producing the predictive picture*"). Ex.1007, 27 ("this method of course requires to estimate the **first order normal derivative** of data at the **boundary**"); Ex.1003, ¶239.

As described in §IV.A.4, above, Saito's techniques represent a natural improvement of image approximation techniques, at least because they can achieve a faster "decay rate" for the error signal at the boundaries of blocks – an improvement that is consistent with the goals of Mualla-Shirani. Ex.1007, 27; Ex.1006, Abstract, 3; Ex.1003, ¶239.

Moreover, Saito's process of PHLCT is described by the '025 patent itself, and the '025 patent cites a subsequent publication by Dr. Naoki Saito that explains the PHLCT method. Ex.1001, Cover, 11:34-12:25; Ex.1003, ¶240.

9. [6.8.DEC]

As discussed above in [6.2.ENC], Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) discloses "*border motion-vector data*" and "*post-quantization*

data.” *See supra*, [6.2.ENC]. In particular, as discussed above, the first stage of Shirani’s two-stage hybrid error concealment technique uses motion vectors of “neighboring blocks” (“***border motion-vector data***”) to estimate motion vectors of subblocks adjacent to the missing block. Ex.1006, 3. Furthermore, as discussed in §IV.A.1, above, the motion vectors of these neighboring blocks (“***border motion-vector data***”) are encoded and transmitted to the decoder. Ex.1005, 71, 74; Ex.1003,, ¶241. Because the encoding and transmission of motion vector information is performed for all macroblocks in a frame/picture, Mualla teaches that the motion vectors of neighboring blocks (“***border motion-vector data***”) around a missing block are encoded and transmitted to the decoder, and can be used by the decoder to perform temporal error concealment in Shirani’s first-stage. Ex.1003, ¶241.

Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) also discloses conventional “***motion vector data***.” *See supra*, §IV.A.1; *see infra*, [6.14.DEC]. As discussed above, Mualla encodes and transmits motion information for each macroblock in a frame, which includes motion information for the macroblock being decoded (“***motion vector data***”) as well as motion information for neighboring blocks (“***border motion-vector data***”) which can be used for error concealment in case the macroblock being decoded is received in error or corrupted. Ex.1005, 71, 74; Ex.1003, ¶242.

In particular, even if the macroblock is detected as having errors, a POSITA would have understood or found obvious that, in the system of Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka), motion information for a macroblock being decoded (“*motion vector data*”) can be received and decoded. Ex.1003, ¶243. For example, Mualla explains that a macroblock is detected as having an “error” if the “**decoded motion vector** points outside the permissible range.” Ex.1005, 232; 244 (explaining that “the motion vector of a damaged block is usually lost or **erroneously received**” and disclosing techniques for recovering a “damaged motion vector”); *see also* Ex.1008, 4 (“Transmission errors can be detected in a variety of ways ... **detect** that the **decoded motion vectors**, DCT coefficients, or quantizer step sizes **exceed their permissible range** (semantic violations)”). In view of these teachings, even if a macroblock is detected as having errors, and even if a particular example disclosed in Mualla or Shirani refers to motion vectors as “missing,” “lost,” or “not available,” a POSITA would have understood or found obvious that the techniques of Mualla and Shirani are applicable to motion vectors (“*motion vector data*”) that are **received** and detected as having errors. Ex.1003, ¶243.

As discussed in [6.1.DEC], above, Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) includes a multiplexer that “demultiplex[es]” received data. In addition, Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka)

implements an entropy decoder to perform entropy decoding on the demultiplexed data. *See supra*, [6.1.DEC]. In particular, Mualla also discloses an “**entropy decoder**” in a “[t]ypical video communication system” that “obtain[s] the output video” (“**generate ... motion-vector data, the post-quantization data, the border motion-vector data and parameter data required for constructing a specific syntax structure**”). Ex.1005, 227-228, FIG. 9.1; Ex.1003, ¶244.

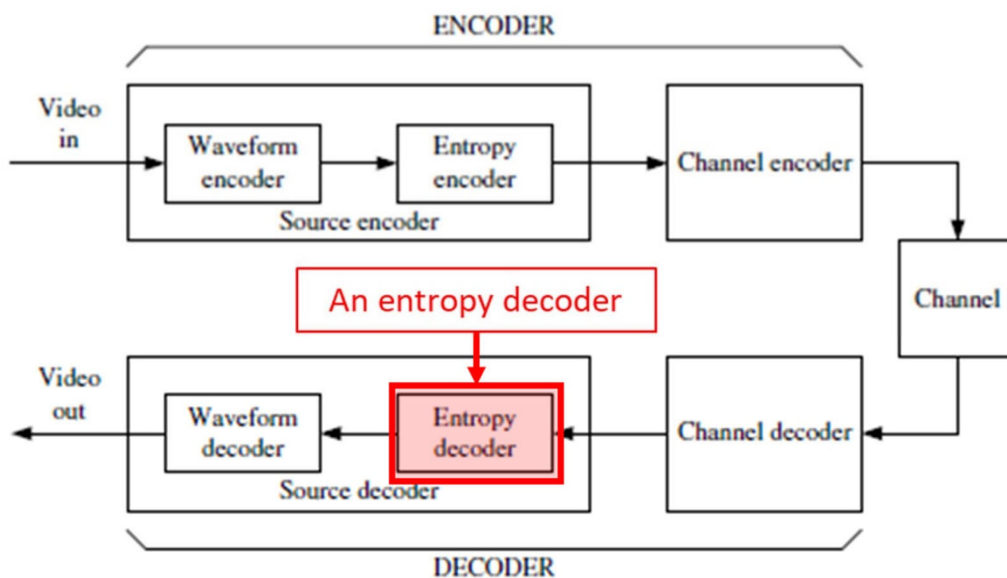


Figure 9.1: Typical video communication system

Ex.1005, FIG. 9.1 (annotated and color added).

10. [6.9.DEC]

Mualla discloses that the H.263 standard includes the techniques of inverse-quantization. Ex.1005, 51-52, 76. Specifically, Mualla discloses that “quantization is achieved by dividing the transform coefficient by a quantization step size and rounding the result (**post-quantization data**)” ... [i]nverse

quantization is the process of reconstructing an approximation of the original coefficient by multiplying the quantized coefficient by the same step size”

(“*inverse-quantization to the post-quantization data to generate post-quantization orthogonal transform coefficients data*”). Ex.1005, 76. Mualla illustrates an “*inverse quantizer*” in its example H.263 video encoder. Ex.1005, FIG. 3.1. Dr. Reader explains that a POSITA would have recognized and found obvious that a similar component would have been included in the decoder of a codec. Ex.1003, ¶245.

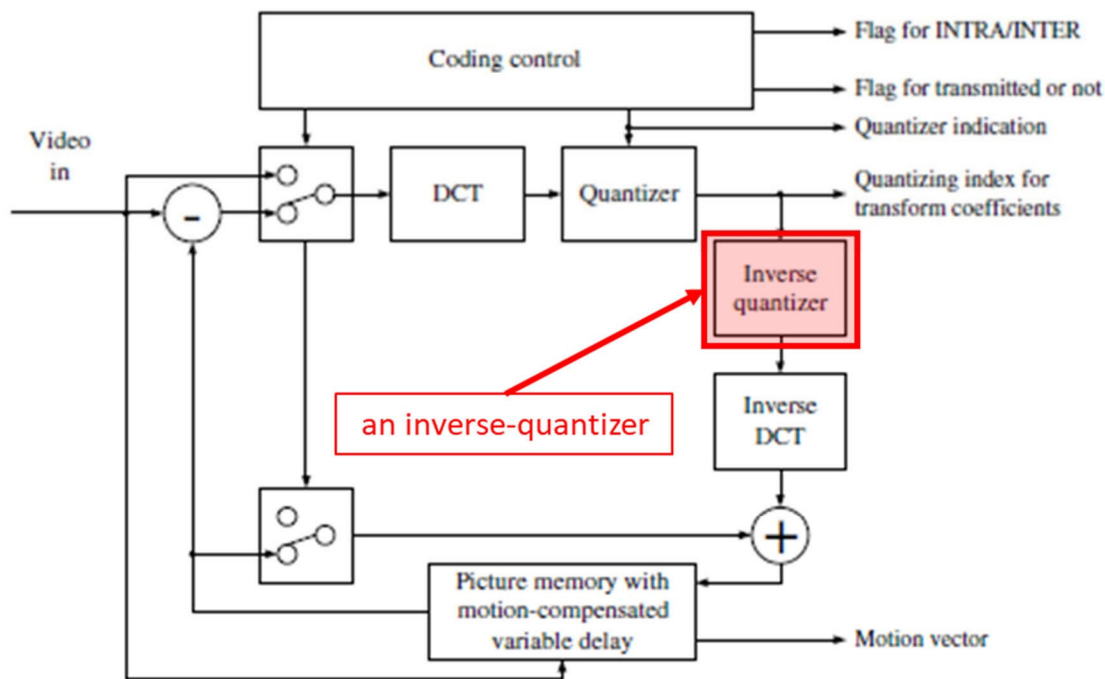


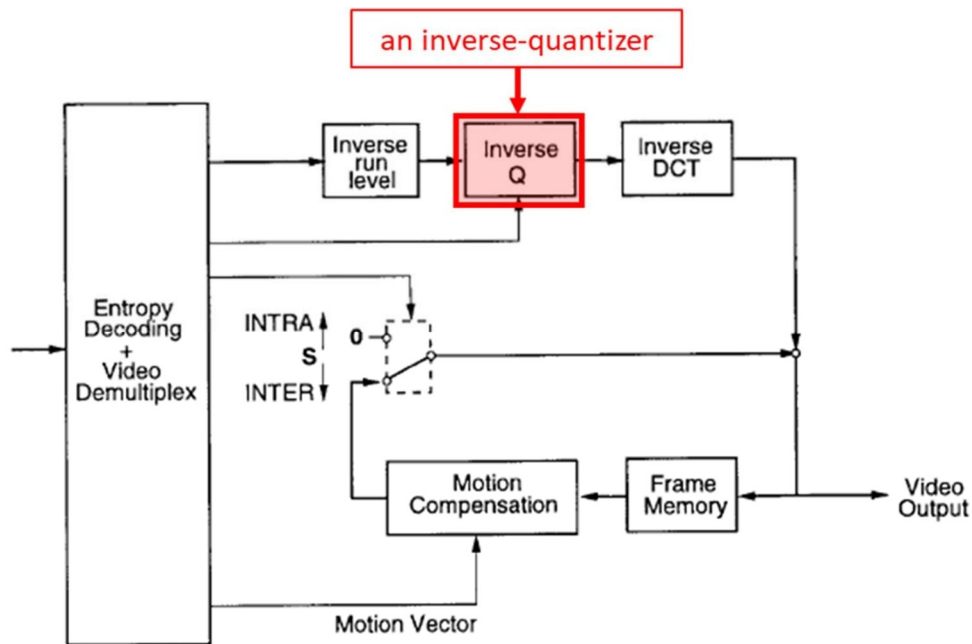
Figure 3.1: H.263 video encoder

Ex.1005, FIG. 3.1 (annotated and color added).

In particular, a POSITA would have recognized or found obvious that the decoder would “*perform inverse quantization,*” for example, by using an “*inverse-*

quantizer,” as this was well-known before the Critical Date. Ex.1003, ¶246.

Indeed, Girod corroborates the use of “inverse-quantizer[s]” in H.263 decoders and illustrates this component in its example H.263 decoder. Ex.1008, 2, FIG. 1(b).



Ex.1008, FIG. 1(b) (annotated and color added).

11. [6.10.DEC]

As discussed in §IV.A.1, above, Mualla discloses an H.263-based system that implements inverse-orthogonal transforms. Ex.1005, 51-52, 76.

Specifically, Mualla discloses the encoder performs “transform coding” where a “unitary space-frequency transform is applied to each block [in an input frame] to produce an $N \times N$ block of transform (spectral) coefficients that are then suitably quantized” (“*post-quantization orthogonal transform coefficients data*”). Ex.1005, 52, FIG. 2.9; Ex.1003, ¶247. At the decoder, “an inverse transform is

applied to reconstruct the frame” (“*an inverse-orthogonal transformer to perform inverse-orthogonal transform to the post-quantization orthogonal transform coefficients data*”). *Id.* FIG. 2.9 of Mualla illustrates an “[i]nverse transform” component in an example “transform coding system.” *Id.*

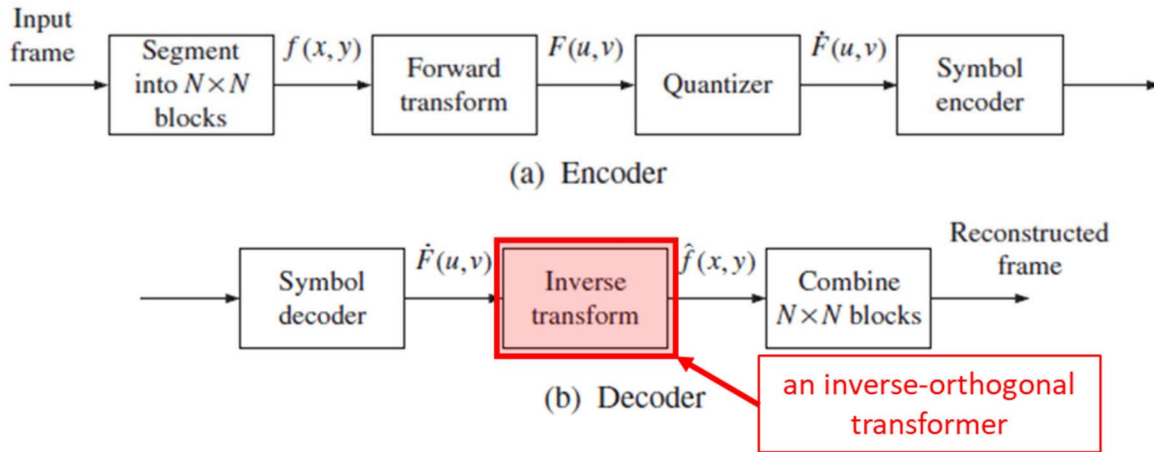
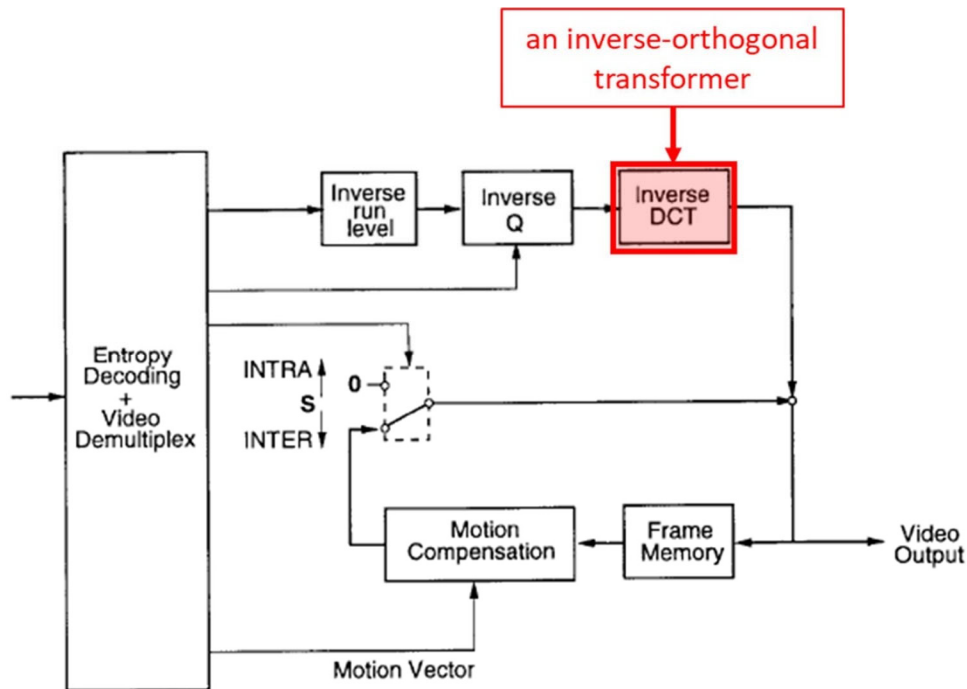


Figure 2.9: Block diagram of a transform coding system

Ex.1005, FIG. 2.9 (annotated and color added).

Moreover, a POSITA would have recognized or found obvious that the decoder would include an “*inverse-orthogonal transformer*” as this was well-known in the art before the Critical Date. Ex.1003, ¶248. Indeed, Girod corroborates the use of inverse transforms in H.263 decoders. Ex.1008, 2, FIG. 1(b); Ex.1003, ¶248.



Ex.1005, FIG. 1(b) (annotated and color added).

12. [6.11.DEC]

As discussed above, the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements a two-stage hybrid error concealment in **both** the **encoder** and the **decoder** by using “interactive techniques,” as taught by Mualla. Ex.1005, 246-250; Ex.1003, ¶249. As described above in [6.2.ENC] and [6.6.ENC]-[6.7.ENC], the **encoder** of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination defines “*a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion vector data*” and “*generate[s] an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson’s Equation, thus producing a first predictive picture.*” See *supra*, [6.2.ENC],

[6.6.ENC]-[6.7.ENC]. Thus, a POSITA would have understood or found obvious that the **decoder** of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination would perform analogous operations by using the same two-stage hybrid error concealment technique. Ex.1003, ¶249.

Furthermore, Mualla refers to the first-stage temporal concealment as “**motion-compensated** error concealment.” Ex.1005, 27. Mualla explains that “[a]s in **motion-compensated prediction**, the process of temporal concealment involves two stages: *concealment displacement estimation* and *displacement compensation*, as shown in Figure 9.8(a). For this reason, temporal concealment is sometimes referred to as **motion-compensated concealment**.” Ex.1005, 243, Figure 9.8(a) (reproduced below, annotated); Ex.1003, ¶250.

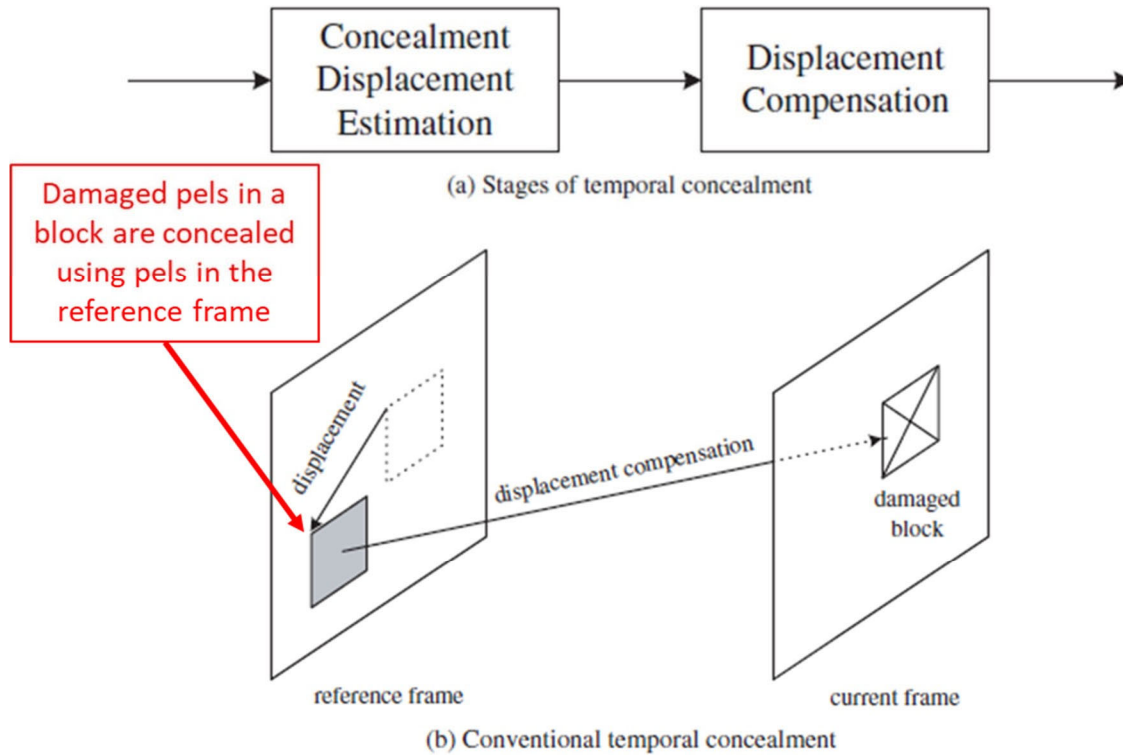


Figure 9.8: Temporal error concealment

Ex.1005, FIG. 9.8 (annotated)

Moreover, as Dr. Reader explains, “the additional techniques of Shirani and Saito are simply ‘algorithms’ that can be executed by computer program modules” (“*a zone-border motion compensator*”). Ex.1006, 1-2; Ex.1007, 6; Ex.1003, ¶251. Accordingly, the encoding features discussed in encoding limitations [6.2.ENC], [6.6.ENC]-[6.7.ENC] would be similarly implemented within the decoder of Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka). Ex.1003, ¶251.

Thus, a POSITA would have understood or found obvious that the **decoder** of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination implements “motion compensated error concealment techniques” as part of a two-

stage hybrid “error concealment” algorithm (the claimed “*zone-border motion compensator*”), which includes a first-stage temporal concealment stage (which defines “*a boundary condition of a border that corresponds to the border motion vector data, from the reference picture based on the border motion vector data*”) and a second-stage spatial concealment stage (which “*generate[s] an estimated video signal in each rectangular zone in the picture to be coded, that satisfies Poisson’s Equation, thus producing a first predictive picture*”). Ex.1003, ¶252.

13. [6.12.DEC]

Mualla describes a well-known decoding process in which a “prediction” (“*the first predictive picture*”) generated by motion-estimation and motion-compensation (the “*zone border motion compensation*” of Mualla-Shirani-Saito or Mualla-Shirani-Saito-Takenaka) is “added to” (or summed with) (“*combine the first predictive picture and the decoded residual picture*”) an “error signal” (“*the decoded residual picture*”), which is the result of inverse DCT, to “reconstruct the current frame” (“*generate a decoded moving picture signal*”). Ex.1005, 27, 76, FIG. 10.8; *see supra* [6.11.DEC]; Ex.1003, ¶253. Indeed, Mualla’s error concealment technique “*generate[s] a decoded moving picture signal*” by concealing missing/erroneous/damaged blocks, as shown in Figure 10.8. *Id.*

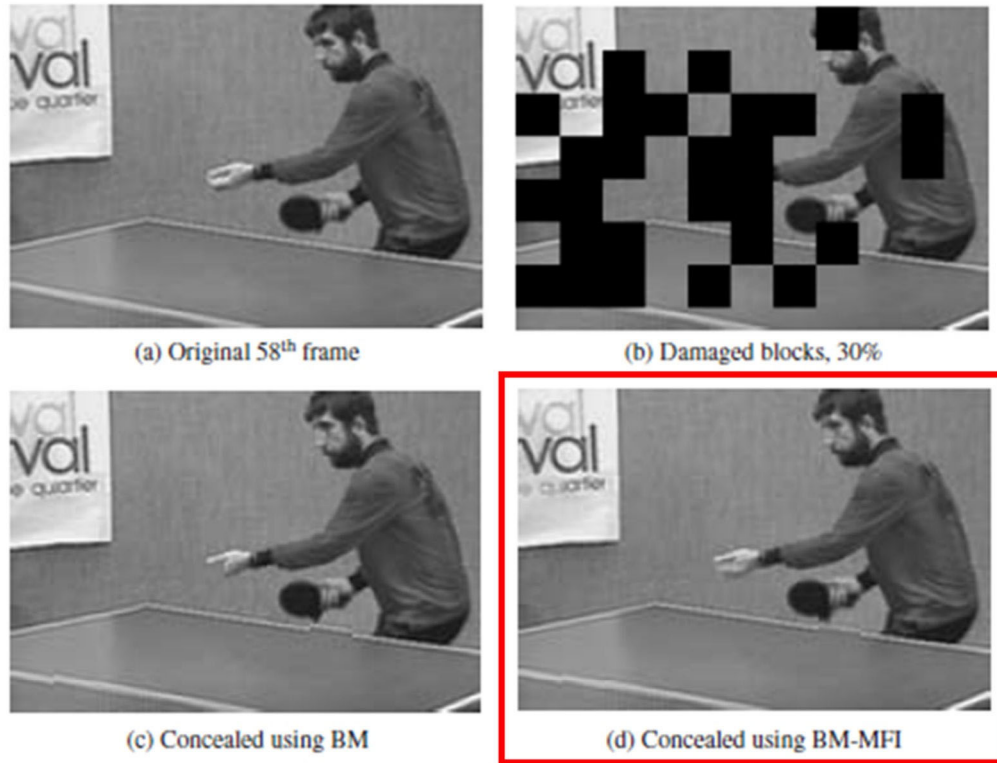


Figure 10.8: Subjective quality of concealed 58th frame of QSIF TABLE TENNIS at 30 frames/s with a block loss rate of 30%

combin[ing] the first predictive picture
and the decoded residual picture to
generate a decoded moving picture signal

Ex.1005, FIG. 10.8 (annotated).

Thus, for a given frame (picture) that is being decoded, (i) coded data for some macroblocks of the frame are received without error and are decoded using Mualla’s conventional “motion estimation” and “motion compensation,” whereas (ii) coded data for other macroblocks of the frame are erroneously received and concealed using the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) two-stage hybrid error concealment technique. The result is an aggregate predicted frame (“*the first predictive picture*”) that is combined with Mualla’s “decoded

[displaced-frame difference] DFD” (“*the decoded residual picture*”). The combining technique is analogous to the combining technique implemented in Mualla’s conventional motion-estimation-compensation decoding. Ex.1005, FIG. 1.1; Ex.1003, ¶¶254-257.

14. [6.13.DEC]

Mualla describes the well-known feature of a decoder including a “*memory*” to store a “*reference picture*” which is the “*decoded moving-picture signal*” that was generated by combining the residual picture and the predictive picture. Ex.1005, 26-27, FIG. 1.1; *see supra*, [6.12.DEC]; Ex.1003, ¶258. For example, Mualla discloses that “reference frame[s]” (“*reference picture[s]*”) are “stored in a frame buffer” (“*a memory*”). *Id.*

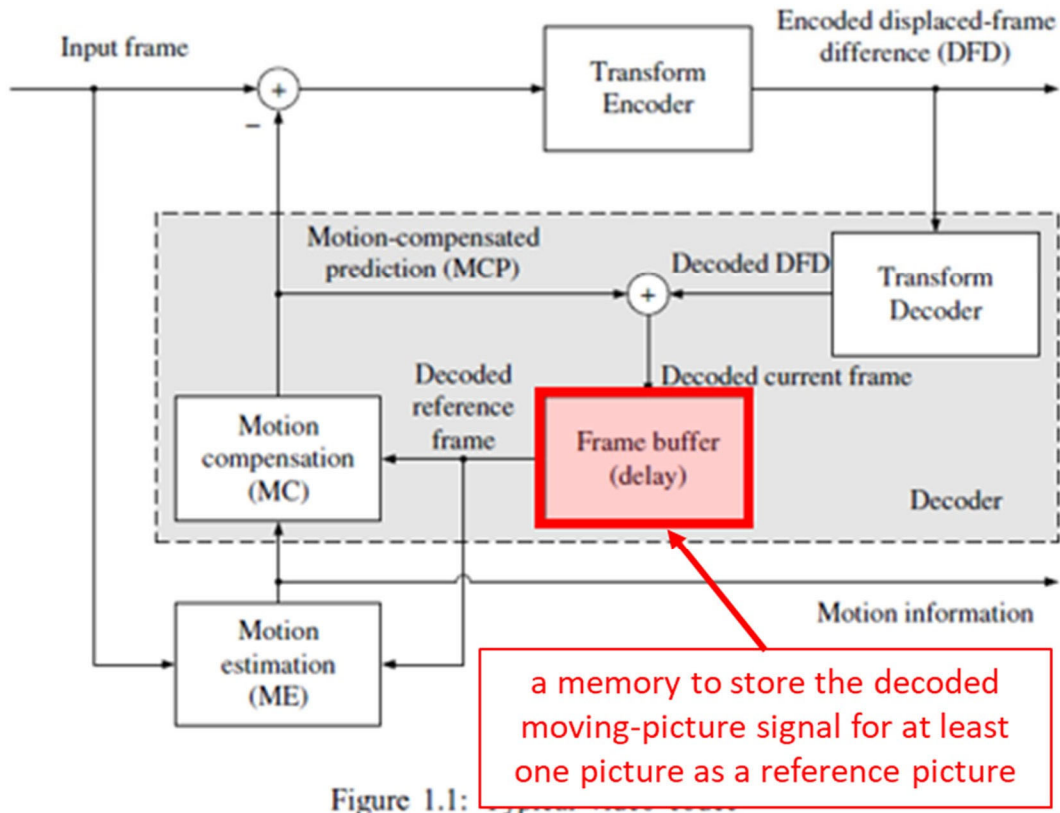
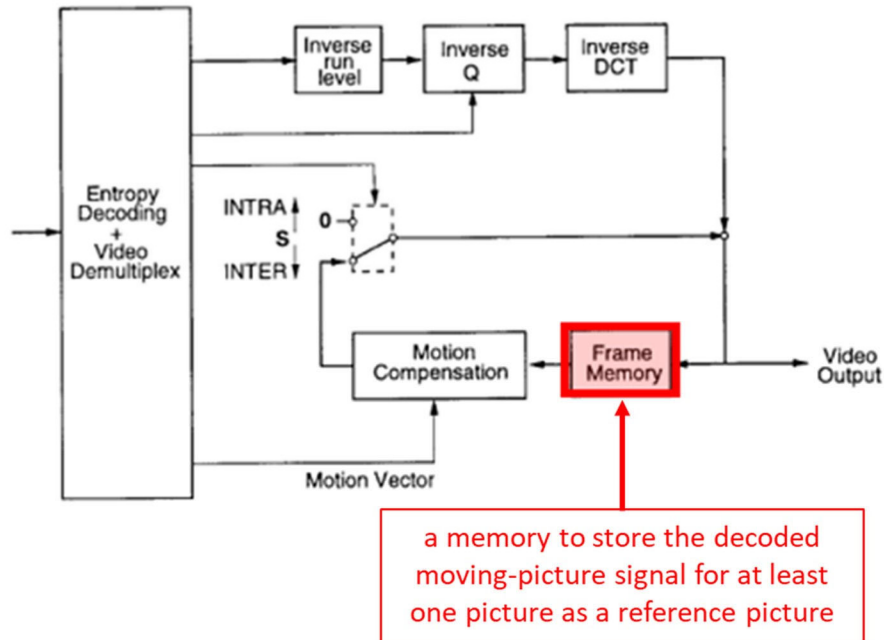


Figure 1.1:

Ex.1005, FIG. 1.1 (annotated and color added).

Additionally, the use of a “*memory*” to “*store ... a reference picture*” was well-known in the art before the Critical Date. Ex.1003, ¶259. For example, Girod discloses that a H.263 decoder includes a “frame memory” which stores “decoded frames.” Ex.1008, 2.



Ex.1008, FIG. 1(b) (annotated and color added).

15. [6.14.DEC]

Mualla's "typical video codec" includes the well-known feature of "motion compensation." Ex.1005, 26-27, 76, 115-116, FIG. 1.1, 2.16; Ex.1003, ¶260. Indeed, Mualla discloses that a "motion estimation (ME) block uses a motion model to estimate the movement that occurred between [a] current frame and a reference frame" ("*specify a corresponding rectangular zone in the reference picture based on the motion vector data*"). Ex.1005, 26-27, 116-117; Ex.1003, ¶260. Then, "[a]t the decoder, the same reference frame is used along with the received motion information to produce the same prediction" ("*thus generating a second predictive picture*"). *Id.* FIG. 1.1 below illustrates motion compensation in Mualla's typical video codec.

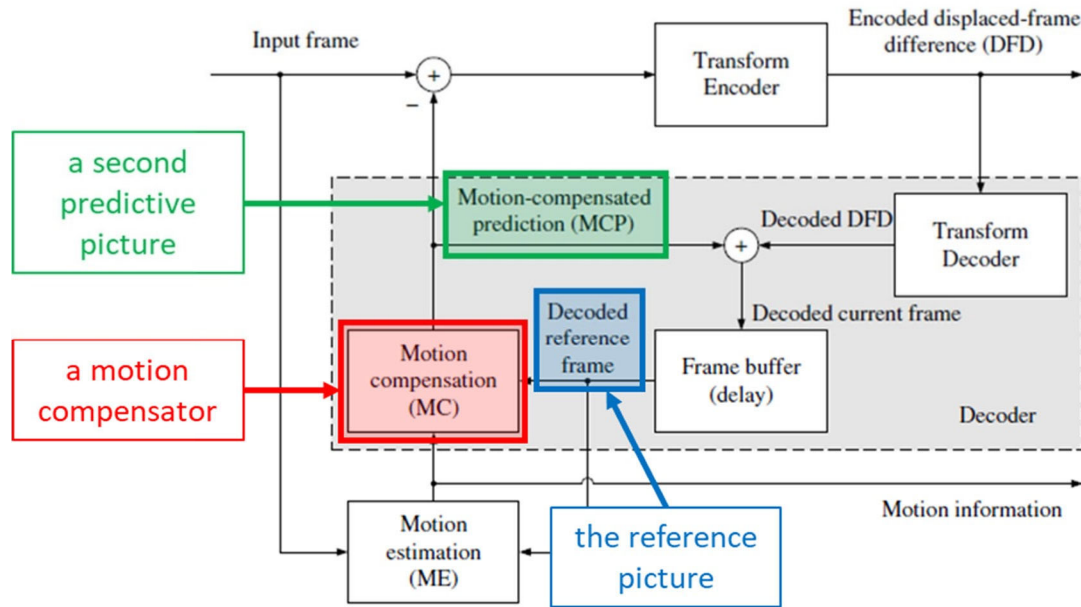


Figure 1.1: Typical video codec

Ex.1005, FIG. 1.1 (annotated and color added).

Additionally, Mualla discloses a process of using a “motion vector” (“*based on the motion vector data*”) to determine a “block” in the reference frame (“*a corresponding rectangular zone in the reference picture*”) that matches the current frame. Ex.1005, 76; Ex.1003, ¶261. This process is illustrated in FIG. 2.16 below.

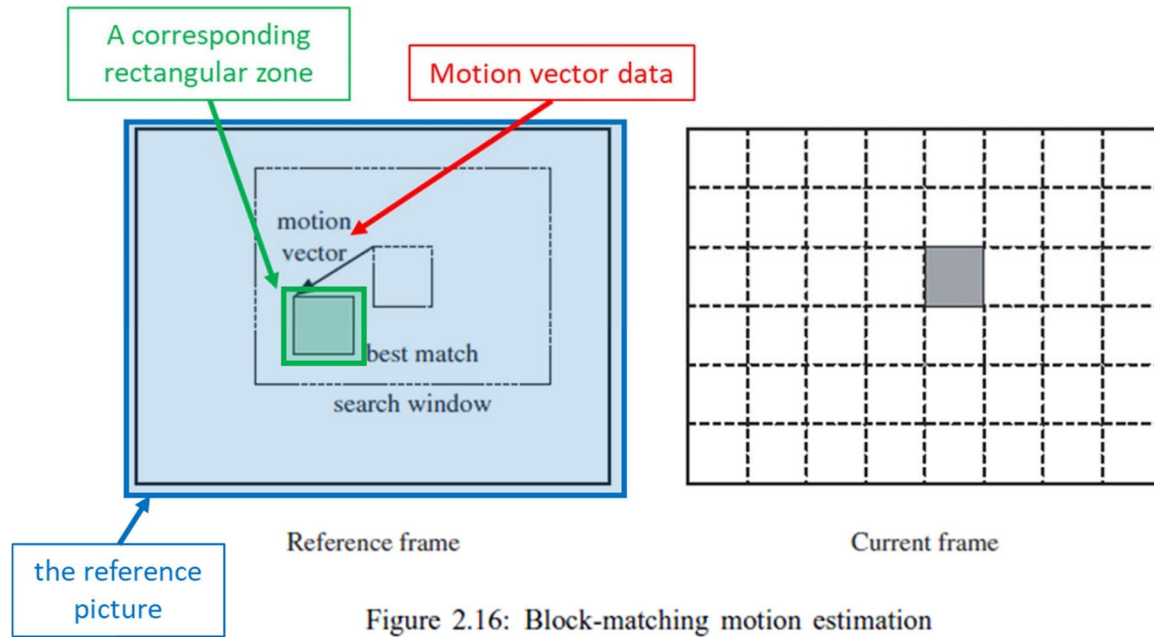


Figure 2.16: Block-matching motion estimation

Ex.1005, FIG. 2.16 (annotated and color added).

Moreover, the techniques of motion compensation were well-known before the Critical Date. Ex.1003, ¶262. Indeed, Shirani describes the use of “motion compensation” (“*a motion compensator*”) and “motion vectors” (“*motion vector data*”) to identify “motion compensated blocks” (“*a corresponding rectangular zone in the reference picture*”). Ex.1006, 1-2.

16. [6.15.DEC]

Mualla discloses that its error concealment techniques can be deployed “[w]hen an error is detected,” and that during this period, the “decoding process is stopped.” Ex.1005, 265. From at least this disclosure, a POSITA would have recognized and found obvious that traditional motion compensation (producing a “*second predictive picture*”) would execute until an error was detected by the

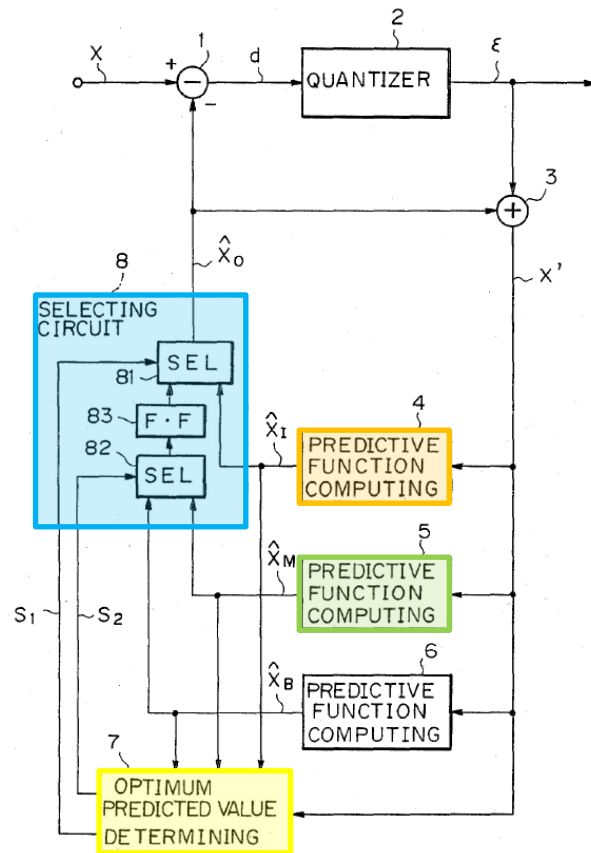
decoder in the bitstream, at which point the error concealment techniques of the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination would be used to generate replacement motion vector and video data (a “*first predictive picture*”). Ex.1005, 265; Ex.1003, ¶263.

As Dr. Reader explains, a POSITA would have recognized or found obvious that switching between the outputs of these two methods of decoding (“*select[ing] either the first predictive picture or the second predictive picture*”) would be effectuated by a switching component, such as a “*selector*.” Ex.1003, ¶264.

Indeed, Girod (cited by Mualla) corroborates selecting between two different decoding modes in response to detected errors, as “an erroneous bit stream cannot be gracefully decoded by an ‘off-the-shelf’ video decoder build [sic] for error-free transmission.” Ex.1008, 4. Girod explains that, in the case of errors, “[s]pecial provisions for error detection, resynchronization, and concealment are required.”

Id. Takenaka also corroborates the use of “selectors” to switch between “predictive function computing circuits,” which a POSITA would have understood includes circuits for traditional motion compensation and error concealment, capable of simultaneously generating two results and selecting between those two results. Ex.1023, 2:52-3:26, 7:67-8:38, Figs. 1, 10; Ex.1003, ¶265.

Fig. 1



Ex.1023, FIG. 1 (annotated and color added)

Moreover, Mualla uses a “*selector*” to control inter or intra coding.

Ex.1005, 72, FIG. 3.1. As Dr. Reader explains, using a selector to switch between normal decoding and error concealment “would have been a natural extension of techniques disclosed within Mualla.” Ex.1003, ¶266. Further, a POSITA would have combined the teachings of Takenaka into the Mualla “selector” to switch between traditional motion compensation and error concealment outputs (which outputs are represented as X_I and X_M of the “predictive function computing” circuits in Figure 1 above). Ex.1003, ¶267.

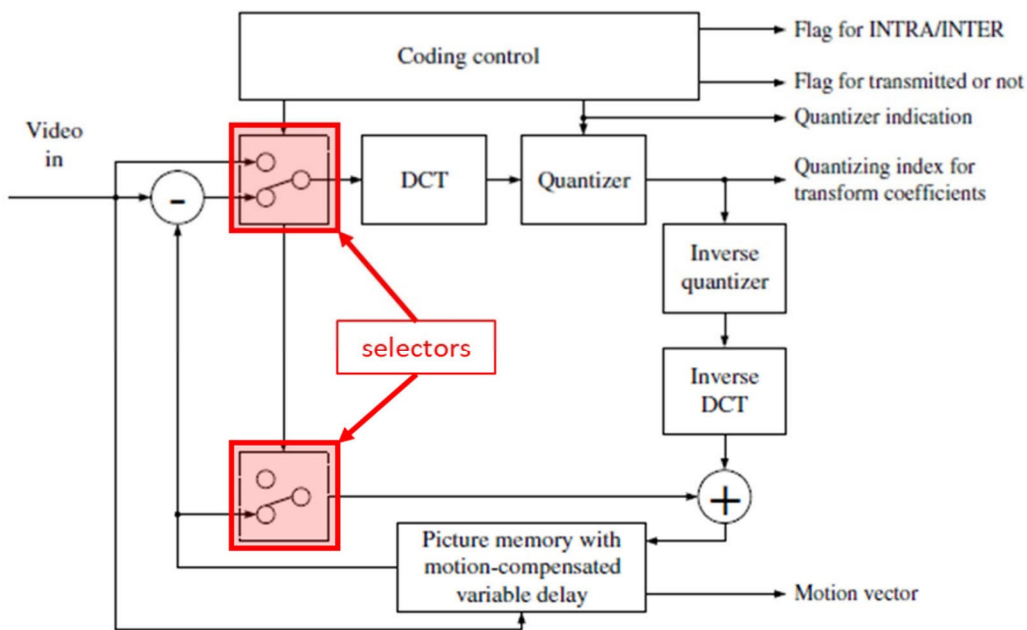
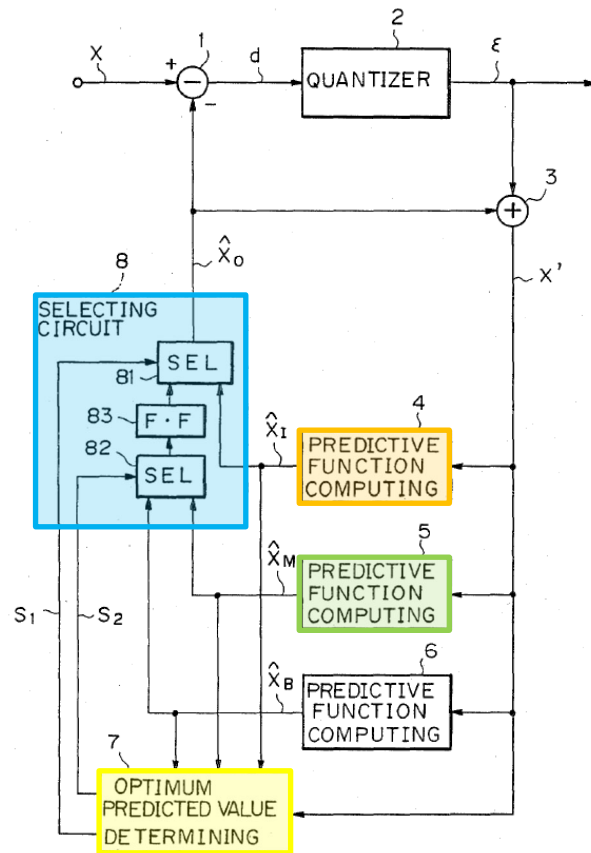


Figure 3.1: H.263 video encoder

Ex.1005, FIG. 3.1 (annotated and color added).

As explained above, Mualla's selectors can be combined with the teachings of Takenaka's "selecting circuit" to select between outputs of different prediction methods, including intra and inter coding or error concealment. See §IV.A.7. And Mualla's "zone-border motion compensator" and "motion compensator" can be modified with Takenaka's "predictive function computing" circuits to produce simultaneous outputs based on different prediction modes, which a POSITA would have understood includes normal coding and error concealment. Ex.1003, ¶268.

Fig. 1



Ex.1023, FIG. 1 (annotated and color added)

17. [6.16.DEC]

As explained above in [6.15.DEC], the Mualla-Shirani-Saito (and Mualla-Shirani-Saito-Takenaka) combination would have switched, via a “selector” or equivalent component, between conventional motion compensation (producing a “*second predictive picture*”) and error concealment (producing a “*first predictive picture*”), depending on the presence of errors in the bitstream. *See supra*, [6.15.DEC]. Additionally, Mualla also discloses that these errors can be detected from “syntax and semantic violations,” the “header information” being one source

of this data (“*decoding control data for decoding control from the parameter data*”). Ex.1005, 231-232. Additionally, Mualla’s H.263 video encoder includes a “coding control” component (“*a decoding controller*”) that is “responsible for varying several parameters to control the rate or the quality of the coded video.” Ex.1005, 72, 75, FIG. 3.1; Ex.1003, ¶269. Examples of functions that are controlled by the coding control component are “INTER/INTRA mode decision at the picture or MB level, the update pattern of the forced INTRA refresh, the TRANSMIT=SKIP decision at the MB level, and the QP and its change at the picture, GOB, or MB level.” Ex.1005, 75; Ex.1003, ¶269. Illustrated below is Mualla’s example H.263 encoder with a “coding control” component that controls operations of the local decoder (the bottom switch).

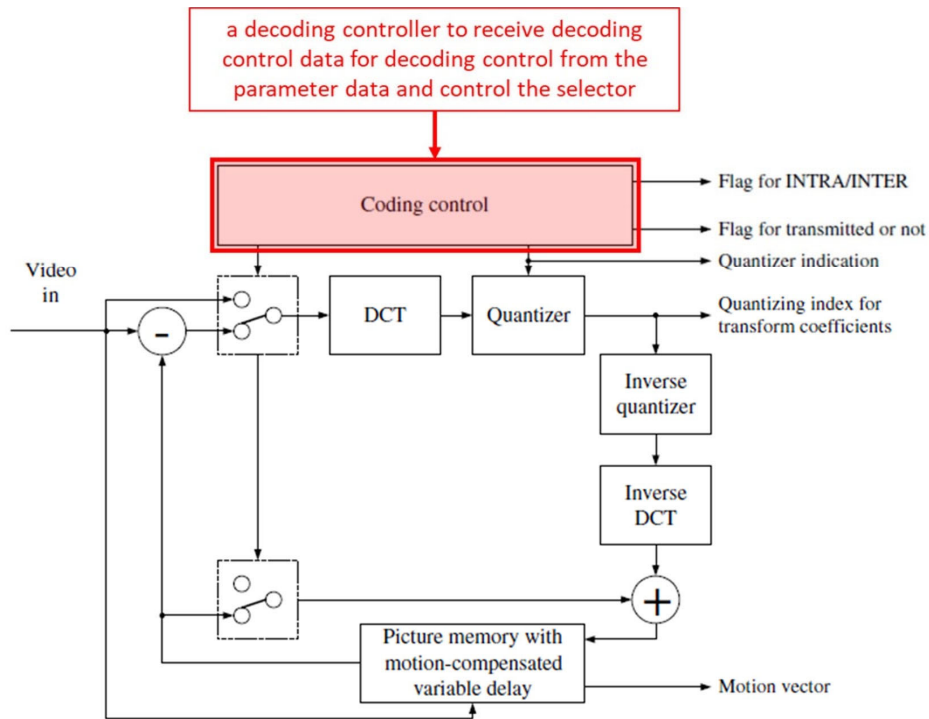


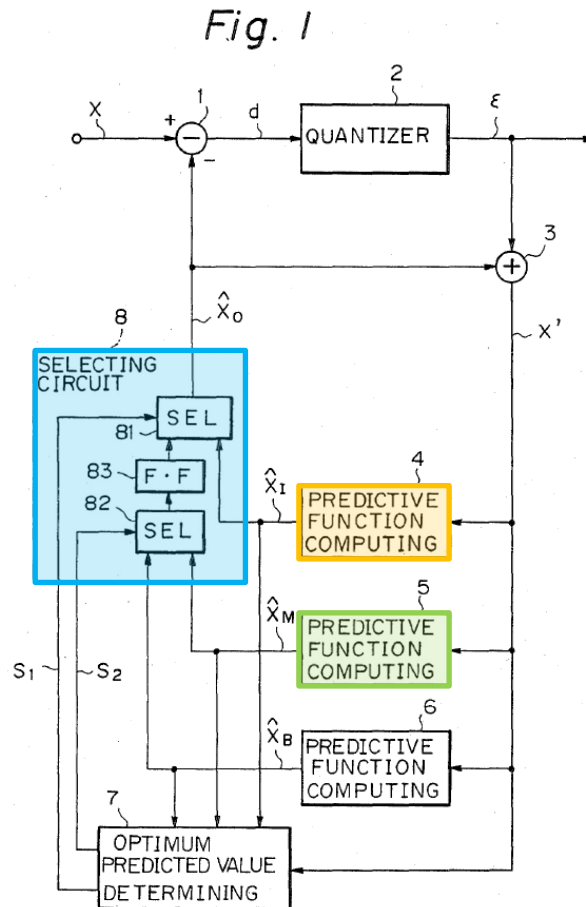
Figure 3.1: H.263 video encoder

Ex.1005, FIG. 3.1 (annotated and color added).

As Dr. Reader explains, a POSITA would have recognized or found obvious that Mualla's decoder would have implemented an analogous coding control component ("*a decoding controller*") which would have "*receive[d] decoding control data for decoding control from the parameter data*" and subsequently, in response to a detected error, "*control[ed] the selector to switch the predictive picture to be supplied to the combiner between the first and second predictive pictures according to the decoding control data.*" Ex.1003, ¶270. Indeed, this function is consistent with the operations of the coding controller described by Mualla, and as Dr. Reader explains, "Mualla's coding controller must necessarily process syntax data (*'decoding control data for decoding control from the*

parameter data') at least because it must make decisions regarding 'inter' or 'intra' prediction." Ex.1003, ¶270.

Additionally, as illustrated in Takenaka's Figure 1 (reproduced below), it was known that a decoder can implement a **decoding controller** that controls a **selector** to switch between the outputs of a first "predictive function computing circuit" (the "**first predictive picture**") and a second "predictive computing circuit" (the "**second predictive picture**"). Ex.1023, FIG. 1; Ex.1003, ¶271.



Ex.1023, FIG. 1 (annotated and color added).

18. [7.pre]

Mualla discloses “software-based video codecs” (“*non-transitory computer readable device*”) that utilize various “algorithms” (“*a computer program comprising a set of instructions*”). Ex.1005, 71, 182, 265, 299. For example, Mualla provides an example of a specific software implementation of H.263 called “Telenor” (citing to a “H.263 TMN software codec” as reference [144]) (“*a method for moving-picture decoding*”). Ex.1005, 265, 299; Ex.1003, ¶272.

Moreover, the use of algorithms in implementing codecs was well-known before the Critical Date. Ex.1003, ¶273. Indeed, Shirani corroborates the use of “algorithms” and “software” in implementing H.263 and H.264 codecs. Ex.1006, 1 (“many of the video compression algorithms (e.g., H.263)”).

19. [7.1.DEC]-[8.3], [10.pre]-[10.13.DEC]

These limitations are obvious as explained above for the corresponding limitations listed in the following table. Ex.1003, ¶274.

Claim	Corresponding Claim
[10.pre]	[6.pre]
[7.1.DEC], [10.1.DEC]	[6.1.DEC]
[7.2.ENC], [10.2.ENC]	[6.2.ENC]
[7.3.ENC], [10.3.ENC]	[6.3.ENC]
[7.4.ENC], [10.4.ENC]	[6.4.ENC]

[7.5.ENC], [10.5.ENC]	[6.5.ENC]
[7.6.ENC], [10.6.ENC]	[6.6.ENC]
[7.7.ENC], [10.7.ENC]	[6.7.ENC]
[7.8.DEC], [8.pre], [10.8.DEC]	[6.8.DEC]
[7.9.DEC], [10.9.DEC]	[6.9.DEC]
[7.10.DEC], [10.10.DEC]	[6.10.DEC]
[7.11.DEC], [10.11.DEC]	[6.11.DEC]
[7.12.DEC], [10.12.DEC]	[6.12.DEC]
[8.1]	[6.14.DEC]
[8.2]	[6.15.DEC]
[8.3]	[6.16.DEC]
[10.13.DEC]	[6.13.DEC]

C. Claims 1-5, 9 (“Moving picture coding”)

1. [1.pre], [4.pre], [9.pre]

Mualla teaches extensive details of an H.263-based video codec system, which implements *coding apparatus, method, and media* for decoding *moving pictures* (e.g., “video”). Ex.1005, 23-26, 31, 36, 51-52, 61-63, 227-228, 241-242, 265. For example, as discussed in §IV.A.1, above, Mualla’s Figure 9.1 and accompanying description disclose a “typical video communication system” which includes a “source encoder” (“*coding apparatus, method, and media*”). Ex.1005,

Figure 9.1 (reproduced below), 227-228. Further details of Mualla’s “source encoder” are illustrated and described with reference to Figures 2.3, 1.1, and 2.9 of Mualla, as explained in §IV.A.1, above. To facilitate understanding, aggregated views of the relation between these different figures of Mualla are shown in Diagrams A and B discussed in §IV.A.1, above, which are reproduced again below. Ex.1003, ¶275.

Additionally, as discussed above in [7.pre], “*non-transitory computer readable device*” is both disclosed by Mualla and further would have been an obvious implementation choice to a POSITA. *See supra* [7.pre]; Ex.1003, ¶276.

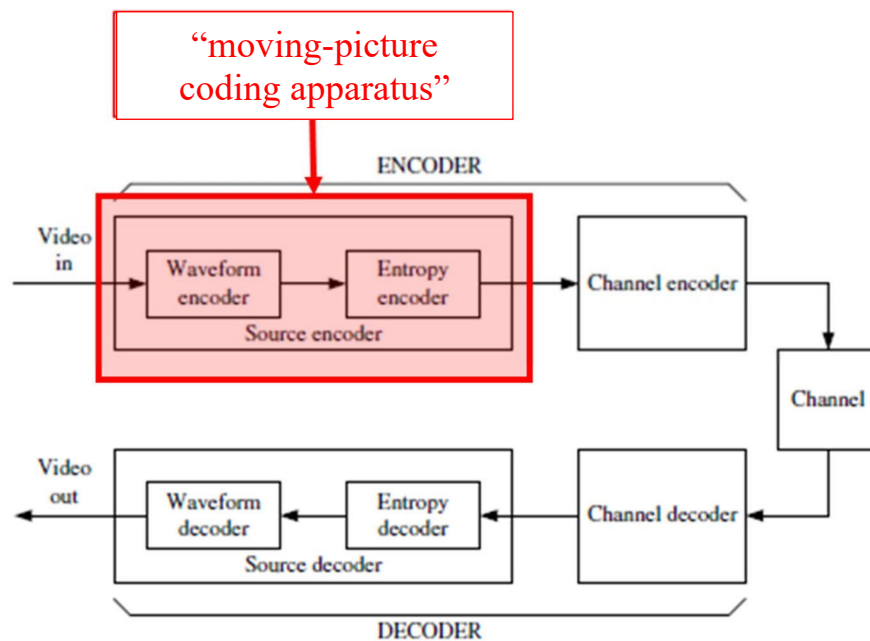


Figure 9.1: Typical video communication system

Ex.1005, FIG. 9.1 (annotated and color added).

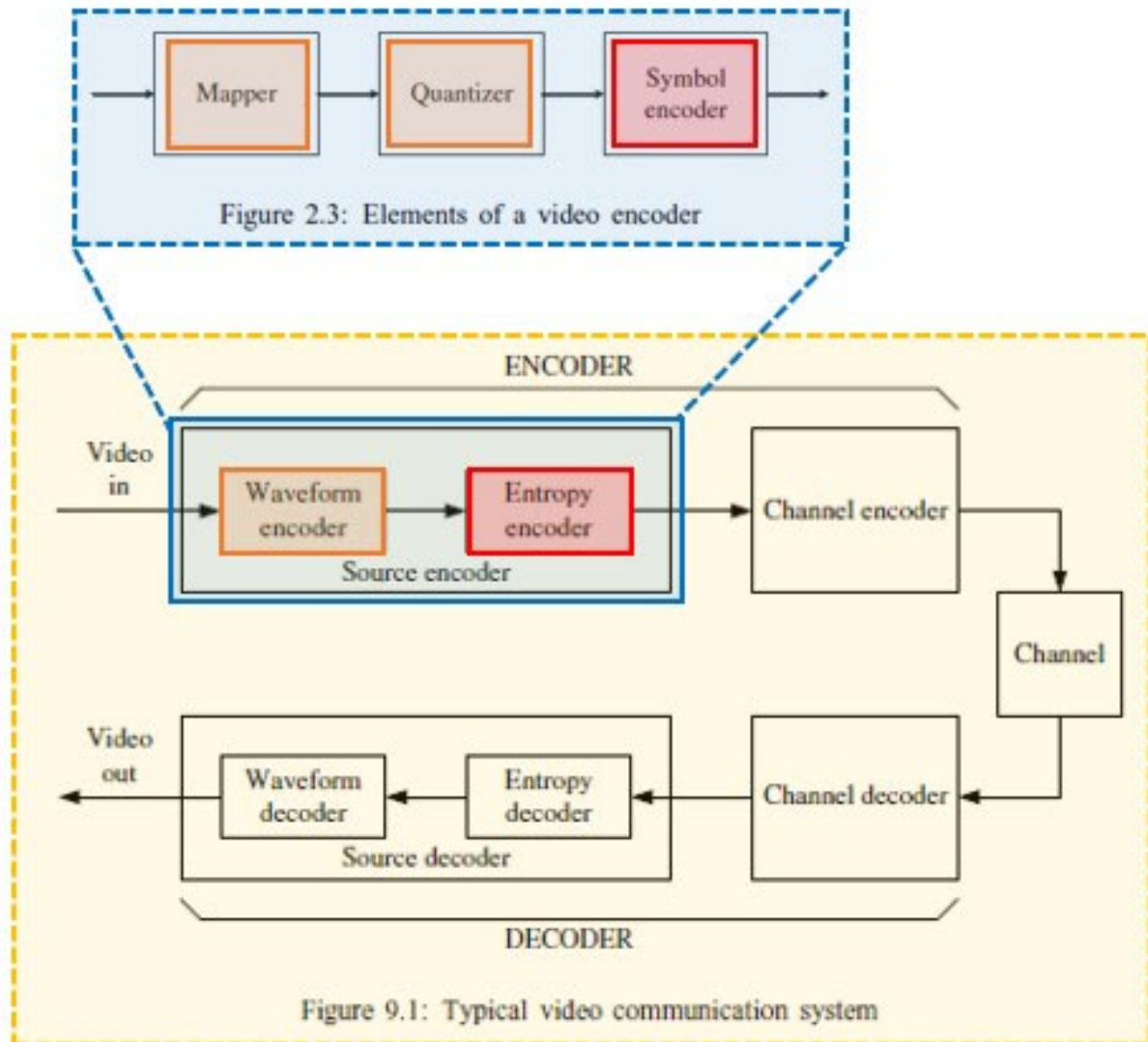


Diagram A (showing the relation between Mualla's Figures 9.1 and 2.3)

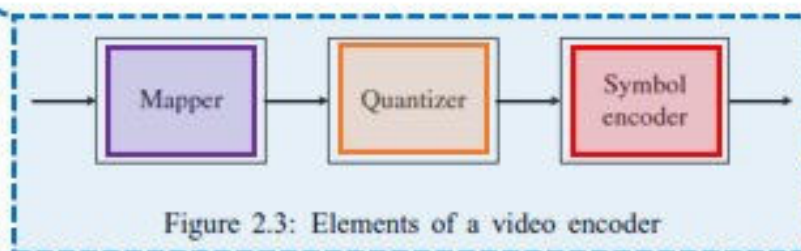
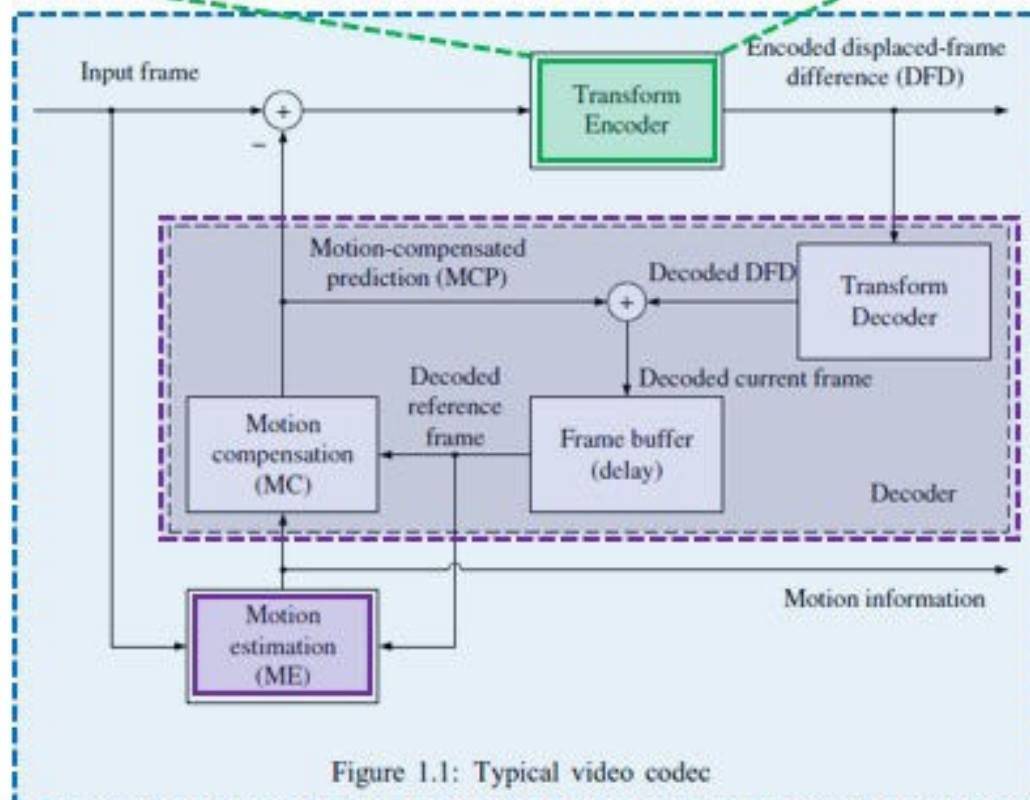
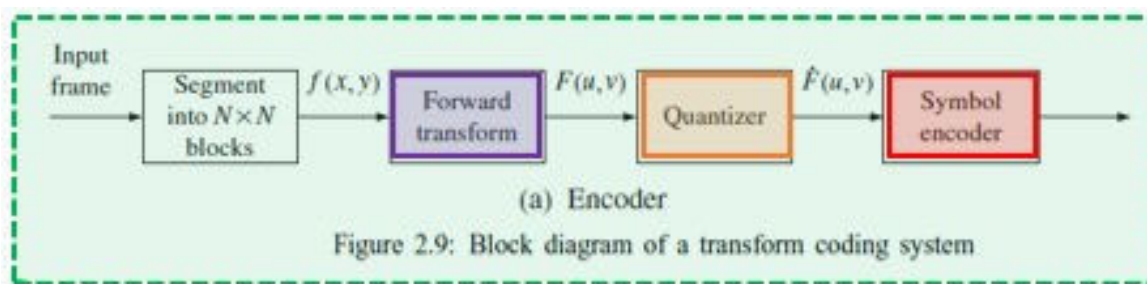


Diagram B (showing the relation between Mualla's Figures 2.3, 1.1, and 2.9)

2. [1.8.ENC], [2.3], [4.8.ENC], [5.3], [9.6.ENC]

Mualla's "typical video codec" illustrates a component ("*a first subtractor*") that subtracts the "[m]otion-compensated prediction (MCP)" ("*the first predictive picture*") from an "[i]nput frame" ("*the picture to be coded*") to produce, after transform encoding, an "[e]ncoded displaced-frame difference (DFD)" ("*a first residual picture*"). Ex.1005, 23-26, FIG. 1.1; Ex.1003, ¶277.

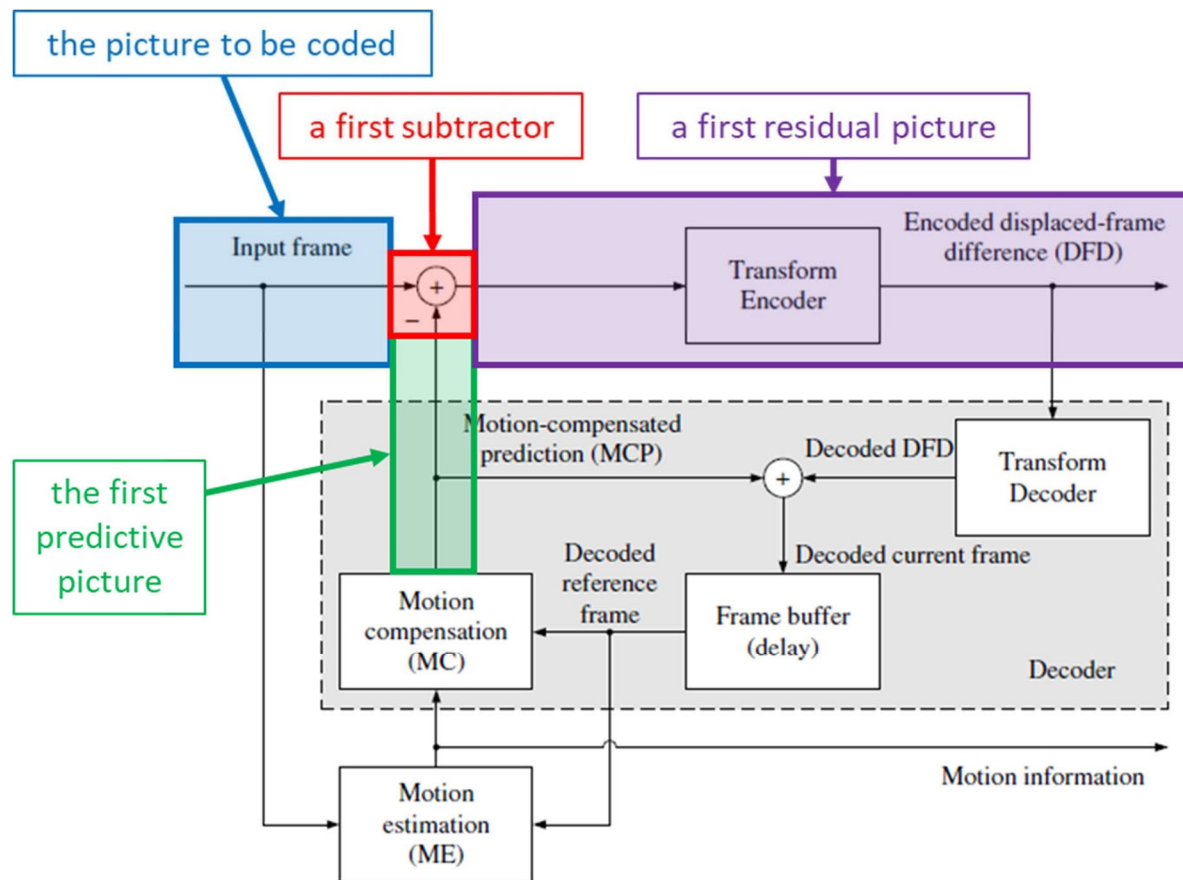


Figure 1.1: Typical video codec

Ex.1005, FIG. 1.1 (annotated and color added).

3. [3.pre]

See [1.pre], [4.pre], [9.pre].

4. [3.1]

Mualla-Shirani-Saito implements an entropy encoder to perform encoding on post-quantization data and border motion-vector data. In particular, Mualla discloses an “*entropy encoder*” in a “typical video communication system,” which corresponds to the “symbol encoder block” shown below. Ex.1005, 205-206; Ex.1003, ¶279.

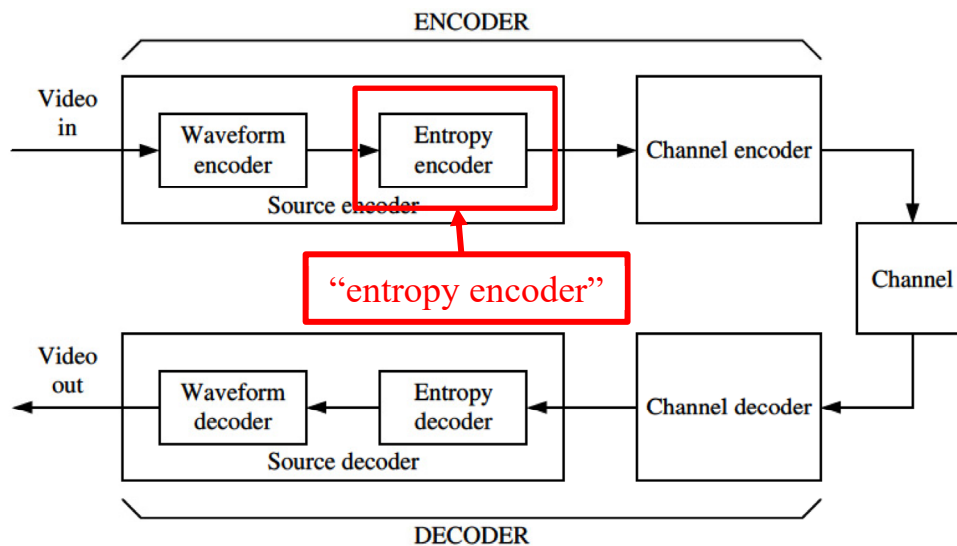


Figure 9.1: Typical video communication system

Ex.1005, FIG. 9.1 (annotated and color added).

As shown in Figure 9.1, Mualla’s “source encoder” receives video input at the “waveform encoder.” *Id.*, 206. The waveform encoder includes “*mapper* and *quantizer* blocks,” and removes “statistical and psychovisual redundancies present in the input video.” *Id.*, 206. Mualla’s mapper “transforms the input raw data into a representation that is designed to reduce statistical redundancy,” *id.*, 18-19, Fig.

2.3, which results in “**border motion-vector data.**” Ex.1003, ¶280⁴. Then the quantizer “reduces the accuracy of the mapper’s output . . . in an attempt to reduce psychovisual redundancy” through “transform coding,” Ex.1005, 18-19, 52, 75, Fig. 2.3, which results in “**post-quantization data.**” Ex.1003, ¶280. Mualla’s entropy encoder (which corresponds to the “symbol encoder block” shown below) performs encoding on the resulting border motion-vector data and post-quantization data to assign “a string of binary bits[] to each symbol at the output of the quantizer” (“*thus generating coded bitstreams*”). Ex.1005, 18-19, Fig. 2.3.

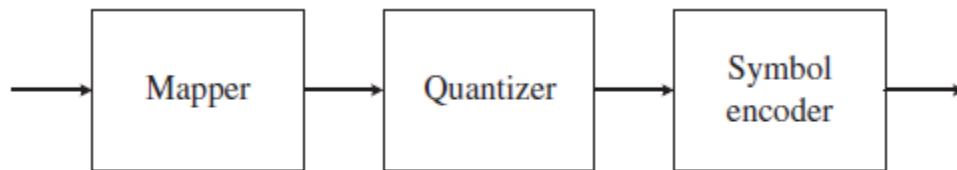


Figure 2.3: Elements of a video encoder

Ex.1005, FIG. 2.3

5. [3.2]

See [6.2.ENC].

⁴ In addition, as described in [6.6.ENC], the Mualla-Shirani combination generates border errors that define the similarity of the neighboring subblock’s motion vector, which is “border motion-vector data.” *See supra*, [6.6.ENC].

6. Remaining limitations of Claims 1, 4, and 9

These limitations are obvious as explained above for the corresponding limitations listed in the following table. Ex.1003, ¶282.

Claim	Corresponding Claim
[1.1.ENC], [4.1.ENC], [9.1.ENC]	[6.3.ENC]
[1.2.ENC], [4.2.ENC], [9.2.ENC]	[6.4.ENC]
[1.3.ENC], [4.3.ENC], [9.3.ENC]	[6.5.ENC]
[1.4.ENC], [1.5.ENC], [4.4.ENC], [4.5.ENC], [9.4.ENC]	[6.6.ENC]
[1.6.ENC], [4.6.ENC], [9.5.ENC]	[6.7.ENC]
[1.7.ENC], [4.7.ENC]	[6.3.ENC]
[1.9.ENC], [1.10.ENC], [4.9.ENC], [4.10.ENC], [4.14.ENC], [9.7.ENC], [9.8.ENC]	[6.2.ENC]
[1.11.DEC], [4.11.DEC], [9.9.DEC]	[6.9.DEC]
[1.12.DEC], [4.12.DEC], [9.10.DEC]	[6.10.DEC]
[4.13.ENC]	[6.8.DEC]
[2.1], [5.1]	[6.2.ENC]
[2.2], [5.2]	[6.14.DEC]
[2.4], [5.4]	[6.15.DEC]
[5.5]	[6.16.DEC]

V. THE BOARD SHOULD REACH THE MERITS OF THIS PETITION

A. *Advanced Bionics* Favors Institution

This Petition presents new prior art and arguments for which the file history contains no indication the Office ever previously considered in connection with the '025 patent. Indeed, none of Mualla, Shirani, Saito, or Takenaka is identified on the face of the '025 patent as having been cited by either the applicant or Examiner during original examination, and certainly were not applied in a rejection or identified in the Examiner's reasons for allowance.

Because the '025 patent's original examination did not involve negotiation between the applicant and the Examiner regarding the prior art teachings applied in the Petition, this Petition's art and arguments cannot be said to be the same as, or substantially similar to, the art or arguments previously presented to the Office in connection with the '025 patent. Although the Examiner cited another reference by Saito (Saito-2005) during prosecution, the Examiner never rejected any claim over Saito-2005, and the prosecution history does not indicate the Examiner addressed or appreciated the teachings within Saito-2005 of Poisson's equation. And, to the extent file history contains any other prior art teachings or arguments cumulative to those presented in this Petition, the Examiner erred in not recognizing the significance of those teachings or arguments in relation to the unpatentability of claims 1-10. *Advanced Bionics LLC v. MED-EL*

Elektromedizinische Gerate GmbH, IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020) (precedential) (citing *Becton, Dickinson and Co. v. B. Braun Melsungen AG*, IPR2017-01586, Paper 8 (PTAB Dec. 15 2017) (precedential)).

Additionally, although the Office previously considered the teachings of Mualla, Shirani, and Saito in instituting IPR2024-00374, a decision on institution is not a final determination on patentability. Further, the Petition does not ask the Board to re-do or modify its consideration of those references; rather, the Petition simply requests review of the '025 patent under much of the same evidence and argument the Board already found to meet the statutory threshold for institution.

Accordingly, Petitioner submits that exercise of the Board's discretion under § 325(d) is unwarranted.

B. *General Plastic* Factors Favor Institution

Under the relevant *General Plastic* factors, the Petition should be instituted. *See General Plastic Industrial Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19 (PTAB Sept. 6, 2017) (precedential); *Valve Corp. v. Electronic Scripting Products, Inc.*, IPR2019-00062, Paper 11 (PTAB Apr. 2, 2019) (precedential).

Petitioner has not previously filed a petition directed to any claim of the '025 patent. Samsung Electronics Co., Ltd. ("Samsung") is the only party to have filed an IPR petition challenging the '025 patent: IPR2024-00374. Here, Petitioner is

neither “the same party” as Samsung, “nor possess[es] a significant relationship” with Samsung, and thus, “under *Valve, General Plastic* factor one necessarily outweighs the other *General Plastic* factors.” *Vivendum Prod. Sols., Inc. v. Rotolight Ltd.*, IPR2023-01218, Paper 12 at 6 (PTAB Apr. 19, 2024).

Additionally, Petitioner “had no involvement with, much less input into, [Samsung’s] IPR.” *Ford Motor Co. v. Neo Wireless LLC*, IPR2023-00763, Paper 28 at 5 (PTAB Mar. 22, 2024). Indeed, Petitioner and Samsung do not have a “significant relationship” and are defendants in “different district court proceedings” involving “different allegedly infringing products.” *Id.* 9-10. To the extent they require consideration, the remaining *General Plastic* factors also weigh in favor of institution.

C. *Fintiv* Factors Favor Institution

Discretionary denial under 35 U.S.C. §314(a) is not appropriate. Patent Owner filed its complaint in the District Court for the Eastern District of Texas on May 10, 2024, then filed two amended complaints on August 2, 2024 and January 17, 2025, respectively.

The district court litigation is at an early stage. Under the current schedule, the claim construction hearing will not occur until September 16, 2025, and expert discovery will not close until November 18, 2025. Ex.1028, 4. The trial is not scheduled to begin until March 2, 2026. Ex.1028, 1. Further, this Petition

challenges all claims of the '025 patent, while only claims 1, 3-4, and 6-10 will be addressed by the district court litigation; thus, this Petition will address the patentability of claims (i.e., claims 2 and 5) that the district court will not address, reducing any overlap between the proceedings. Finally, the merits of this petition are strong, as the Board has already found a reasonable likelihood of prevailing as to certain challenged claims on similar evidence in IPR2024-00374. *See Samsung Elecs. Co., Ltd. v. Advanced Coding Techs. LLC*, IPR2024-00374, Paper 10 at 14 (PTAB Jul. 22, 2024).

Petitioner reserves the right to file an opposition to Patent Owner's discretionary arguments submitted pursuant to the Acting Director's Interim Processes for PTAB Workload Management of March 26, 2025.

VI. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8(a)(1)

A. Real Party-In-Interest

Google LLC is the petitioner and real party-in-interest.⁵

⁵ Google LLC is a subsidiary of XXVI Holdings Inc., which is a subsidiary of Alphabet Inc. XXVI Holdings Inc. and Alphabet Inc. are not real parties-in-interest to this proceeding.

B. Related Matters

The '025 patent is at issue in *Advanced Coding Technologies LLC v. Google LLC*, No. 2:24-cv-00353-JRG (E.D. Tex.) and *Advanced Coding Technologies LLC v. Apple Inc.*, No. 2:24-cv-00572-JRG (E.D. Tex.).

Petitioner is not aware of any other disclaimers, reexamination certificates, or IPR petitions addressing the '025 patent.

C. Grounds for Standing

Petitioner certifies that the '025 patent is available for IPR and Petitioner is not barred from requesting same.

D. Counsel and Service Information

Pursuant to 37 C.F.R. §§ 42.8(b)(3), 42.8(b)(4), and 42.10(a), Petitioner designates the following lead counsel:

- Raghav Bajaj (Reg. No. 66,630), raghav.bajaj@lw.com, Latham & Watkins LLP; 300 Colorado Ave. Suite 2400; Austin TX 78701; 737.910.7370.

Petitioner also designates the following backup counsel:

- Doug Lumish (*pro hac vice* motion to be filed), doug.lumish@lw.com, Latham & Watkins LLP; 140 Scott Drive; Menlo Park, CA 94025; 650.463.2633.

- Patricia Young (*pro hac vice* motion to be filed), patricia.young@lw.com, Latham & Watkins LLP; 1271 Avenue of the Americas; New York, NY 10020; 212.906.1661.
- Linfong Tzeng (Reg. No. 65,140), linfong.tzeng@lw.com, Latham & Watkins LLP; 140 Scott Drive; Menlo Park, CA 94025; 650.463.3092
- Joseph H. Lee (*pro hac vice* motion to be filed), joseph.lee@lw.com, Latham & Watkins LLP; 650 Town Center Drive, 20th Floor; Costa Mesa, CA 92626; 714.755.8046.

Pursuant to 37 C.F.R. § 42.10(b), a Power of Attorney from Petitioner is attached. Petitioner consents to electronic service.

E. IPR Fee

The Director is authorized to charge the fee specified by 37 C.F.R. § 42.15(a) to Deposit Account No. 506269.

VII. CONCLUSION

For the reasons set forth above, Petitioner Google respectfully requests *inter partes* review of the Challenged Claims of the '025 patent.

Respectfully submitted,

Dated: May 12, 2025

By: / Raghav Bajaj /

Raghav Bajaj (Reg. No. 66,630)
raghav.bajaj@lw.com
Latham & Watkins LLP
300 Colorado Ave. Suite 2400
Austin, TX 78701
Telephone: 737.910.7370
Fax: 737.910.7301

Counsel for Petitioner
Google LLC

CERTIFICATE OF COMPLIANCE WITH 37 C.F.R. § 42.24

I hereby certify that this Petition complies with the word count limitation of 37 C.F.R. § 42.24(a)(1)(i) because the Petition contains a total of 13,945 words, as calculated by Microsoft Word's word-count feature. This total excludes the cover page, signature block, and the parts of the Petition exempted by 37 C.F.R. § 42.24(a)(1).

By: / Raghav Bajaj /

Raghav Bajaj (Reg. No. 66,630)
raghav.bajaj@lw.com
Latham & Watkins LLP
300 Colorado Ave. Suite 2400
Austin, TX 78701
Telephone: 737.910.7370
Fax: 737.910.7301

Counsel for Petitioner
Google LLC

CERTIFICATE OF SERVICE

The undersigned certifies that a complete copy of this Petition for *Inter Partes* Review of U.S. Patent No. 8,090,025 and all Exhibits and other documents filed together with this Petition were served on the official correspondence address for the patent shown in Patent Center:

Fabricant LLP
411 Theodore Fremd Road
Suite 206 South
Rye, NY 10580

via FEDERAL EXPRESS next business day delivery, on May 12, 2025.

By: / Raghav Bajaj /

Raghav Bajaj (Reg. No. 66,630)
raghav.bajaj@lw.com
Latham & Watkins LLP
300 Colorado Ave. Suite 2400
Austin, TX 78701
Telephone: 737.910.7370
Fax: 737.910.7301

Counsel for Petitioner
Google LLC