UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC. and AMAZON.COM SERVICES LLC,
Petitioner

v.

INTERDIGITAL, INC.,
Patent Owner.

IPR2026-00195
U.S. Patent No. 10,250,877

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 10,250,877

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

TABLE OF CONTENTS

EXHIBIT LIST

No.	Exhibit
1001	U.S. Patent No. 10,250,877 ("'877 Patent")
1002	File History of the '877 Patent
1003	Declaration of Dr. Joseph P. Havlicek in Support of Petition
1004	Dr. Joseph P. Havlicek CV
1005	Declaration of Dr. Clifford Reader
1006	Dr. Clifford Reader CV
1007	U.S. Patent No. 5,805,222 ("Nakagawa")
1008	U.S. Application No. 2004/0150747 ("Sita")
1009	Thomas Wiegand et al., *WD3: Working Draft 3 of High-Efficiency Video Coding*, JCTVC-E603 v.8 (June 27, 2011) ("Wiegand")
1010	Thomas Davies, *Resolution switching for coding efficiency and resilience*, JCTVC-F158 v.4 (July 7, 2011) ("Davies")
1011	U.S. Patent No. 6,628,714 ("Fimoff")
1012	Joint Chart Setting Forth Parties' Post-Hearing Claim Constructions, *In the Matter of Certain Electronic Devices, Including Smartphones, Computers, Tablet Computers, and Components Thereof*, Inv. No. 337-TA-1373 (Feb. 21, 2024)
1013	Kiran Misra, *AHG18/21: Absolute signaling for resolution switching*, JCTVC-G715 v.3 (Nov. 19, 2011) ("Misra")
1014	U.S. Patent Application No. 2008/0089417 ("Bao")
1015	International Patent Application No. WO 2011/127961 ("Siekmann")
1016	International Patent Application No. WO 1990/00780 ("Dolazza")

No.	Exhibit
1017	U.S. Patent No. 8,199,812 ("Ye")
1018	Alex Wiltshire, *How many frames per second can the human eye really see?*, PCGamer (Jan. 19, 2017), www.pcgamer.com/how-many-frames-per-second-can-the-human-eye-really-see/
1019	Noah Snavely, *Lecture 3: Image Resampling*, Cornell (Spring 2011), www.cs.cornell.edu/courses/cs6670/2011sp/lectures/lec03_resample.pdf
1020	BDTi, *How video compression works*, EETimes (Aug. 6, 2007), https://www.eetimes.com/how-video-compression-works/
1021	Gary J. Sullivan & Thomas Wiegand, *Video Compression—From Concepts to the H.264/AVC Standard*, 93 Proceedings of the IEEE 18 (Jan. 2005)
1022	Niels Laukens, *Adaptive Streaming – a brief tutorial*, EBU Tech. Rev. (Q1 2011)
1023	Real, *Infographic – The History of Digital Video File Formats* (Apr. 22, 2012), https://blog.real.com/digital-video-file-formats/
1024	Gavin Wright, *Definition: interlaced display*, TechTarget, https://www.techtarget.com/whatis/definition/interlaced-display
1025	A. Tamhankar & K. R. Rao, *An Overview of H.264/MPEG-4 Part 10*, 4th EURASIP Conf., Zagreb, Croatia (July 2-5, 2003)
1026	Jordi Ribas-Corbera, et al., *A Generalized Hypothetical Reference Decoder for H.264/AVC*, 13 IEEE Trans. on Circuits and Sys. For Video Tech., 674 (July 2003)
1027	Colt McAnlis & Aleks Haecky, *Understanding Compression, Chapter 4: Variable-Length Codes*, O'Reilly (July 2016), https://www.oreilly.com/library/view/understanding-compression/9781491961520/ch04.html

No.	Exhibit
1028	David A. Huffman, *A Method for the Construction of Minimum-Redundancy Codes*, Proc. of the I.R.E., 1098 (Sept. 1952)
1029	Tino von Roden, *H.261 and MPEG1 – A Comparison*, Conf. Proc. of the 1996 IEEE Fifteenth Annual Int'l Phoenix Conf. on Comp. and Comm'cns, 65 (1996)
1030	*JCT-VC – Joint Collaborative Team on Video Coding*, Int'l Telecomm. Union, https://www.itu.int/en/ITU-T/studygroups/2013-2016/16/Pages/video/jctvc.aspx
1031	*Terms of Reference of the Joint Collaborative Team on Video Coding Standard Development*, JCT-VC (Jan. 2010)
1032	A.W. Johnson & J. Princen, *Drift Minimization in Frequency Scalable Coders Using Block Based Filtering*, IEEE Workshop on Visual Signal Processing and Comm'cn, 231 (September 1993)
1033	Shuai Hu et al., *Optimization of Memory Allocation for H.264/AVC Video Decoder on Digital Signal Processors*, 2008 Congress on Image and Signal Processing
1034	Paige Albiniak, *HDTV: Launched and Counting*, Highbeam Research (Nov. 2, 1998), http://www.highbeam.com/doc/1G1-53190401.html, WayBack Machine (Sept. 24, 2014)
1035	Siwei Ma, *Rate Control for JVT Video Coding Scheme with HRD Considerations*, Proceedings 2003 Int'l Conf. on Image Processing (Nov. 24, 2003)
1036	*All Meetings*, JCT-VC, phenix.int-evry.fr/jct/doc_end_user/all_meeting.php
1037	*Special Section on the Joint Call for Proposals on High Efficiency Video Coding (HEVC) Standardization*, 20 IEEE Trans. on Circuits and Sys. For Video Tech. 1661 (Dec. 2010)
1038	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the first meeting of the Joint Collaborative Team on Video Coding (JCT-*

No.	Exhibit
	VC), Dresden, DE, 15-23 April, 2010, JCT-VC-A200 (Mar. 19, 2012)
1039	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the second meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Geneva, CH, 21-28 July, 2010*, JCT-VC-B200 (Mar. 19, 2012)
1040	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the third meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Guangzhou, CN, 7–15 October, 2010*, JCT-VC-C400 (Jan. 16, 2011)
1041	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the fourth meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Daegu, KR, 20-28 January, 2011*, JCT-VC-D500, v.2 (Apr. 15, 2011)
1042	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the fifth meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Geneva, CH, 16-23 March 2011*, JCT-VC-E600, v.3 (July 19, 2011)
1043	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the sixth meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Torino, IT, 14–22 July 2011*, JCT-VC-F800 (Nov. 18, 2011)
1044	Gary J. Sullivan & Jens-Rainer Ohm, *Meeting report of the seventh meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Geneva, CH, 21–30 Nov. 2011*, JCT-VC-G1100 (Jan. 31, 2012)
1045	Document Information, JCTVC-E603, phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=2471
1046	*About ITU*, https://www.itu.int/en/about/Pages/default.aspx
1047	European Patent Application No. 2,533,537

No.	Exhibit
1048	Japanese Patent Application No. 2013-12887
1049	U.S. Patent Application No. 2012/0257702
1050	Document Information, JCTVC-F158, phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=2623
1051	*ITU-T In Brief*, https://www.itu.int/en/ITU-T/about/Pages/default.aspx
1052	*About MPEG*, https://www.mpeg.org/about-mpeg/
1053	Declaration of Benjamin Bross, *Samsung Elecs. Co., Ltd. v. M&K Holdings Inc.*, IPR2018-00696, Ex. 1002 (Excerpted)
1054	"Resolution," Guangzhou Meeting (C) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1055	"Deblock," Daegu Meeting (D) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1056	"Resolution," Daegu Meeting (D) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1057	Geneva Meeting (E) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php?id_meeting=148&search_category=m&search_id_group=1&search_sub_group=1
1058	"Resolution," Geneva Meeting (E) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1059	"Working Draft," Geneva Meeting (E) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1060	Torino Meeting (F) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php?id_meeting=149&search_id_group=1&search_sub_group=1

No.	Exhibit
1061	“Resolution,” Torino Meeting (F) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1062	“Resolution,” Geneva Meeting (G) – Document Register, phenix.it-sudparis.eu/jct/doc_end_user/current_meeting.php
1063	Mehmet Umut Demircin, et al., *TE2: Compressed Reference Frame Buffers (CRFB), Geneva, CH, 21–28 July, 2010*, JCT-VC-B089 (Mar. 19, 2012)
1064	Zuhang Jia and Qingjiu Huang, *Image Interpolation with Regional Gradient Estimation*, Applied Sci. (July 22, 2022)
1065	Canon, *EOS Movie Compression Options: All-I and IPB*, https://www.canon.com.hk/cpx/en/technical/va_EOS_Movie_Compression_Options_All_I_and_IPB.html
1066	Scott Sargent, *AV Workship’s HD Upres Video Transfer Add On* (May 12, 2021), https://av-workshop.com/av-workshop-video-transfer-hd-upres/
1067	*Interdigital v. Lenovo*, Evidentiary Hearing Transcript (Aug. 14, 2024)

LISTING OF CHALLENGED CLAIMS

Reference	Limitation
Claim 1	
1[pre]	A decoding method of a binary stream to reconstruct a current block of a current image from a reference block of a reference image reconstructed at a different size from the size of said current image, said reconstructed reference image being stored in a decoded picture buffer comprising:
1[a]	motion compensating said reference block of said reconstructed reference image by applying a single horizontal filter GFH and a single vertical filter GFv successively on the lines and on the columns of pixels of said reference block,
1[b]	decoding, for the current block, a residue block, and
1[c]	reconstructing the current block from said residue block and from said motion compensated reference block,
1[d]	wherein said single vertical filter GFv applied on a pixel s is such that GFv(s)=MCIFv(SCFv(s)), where MCIFv is a vertical motion compensation interpolation filter and SCFv is a vertical resampling filter, MCIFv and SCFv being applied jointly and
1[e]	wherein said single horizontal filter GFH applied on a pixel u is such that GFH(u)=MCIFH(SCFH(u)), where MCIFH is a horizontal motion compensation interpolation filter and SCFH is a horizontal resampling filter, MCIFH and SCFH being applied jointly and
1[f]	wherein no resampled version of said reconstructed reference image is stored in the decoded picture buffer.
Claim 4	
4[pre]	A coding method of a current block of a current image from a reference block of a reference image reconstructed at a different size from the size of said current image, said reconstructed reference image being stored in a decoded picture buffer comprising:
4[a]	motion compensating said reference block of said reconstructed reference image by applying a single horizontal filter GFH and a

Reference	Limitation
	single vertical filter GFv successively on the lines and on the columns of pixels of said reference block,
4[b]	calculating, for the current block, a residue block from said current block and from said motion compensated reference block, and
4[c]	coding the residue block in a binary stream,
4[d]	wherein said single vertical filter GFv applied on a pixel s is such that GFv(s)=MCIFv(SCFv(s)), where MCIFv is a vertical motion compensation interpolation filter and SCFv is a vertical resampling filter, MCIFv and SCFv being applied jointly and
4[e]	wherein said single horizontal filter GFH applied on a pixel u is such that GFH(u)=MCIFH(SCFH(u)), where MCIFH is a horizontal motion compensation interpolation filter and SCFH is a horizontal resampling filter, MCIFH and SCFH being applied jointly and
4[f]	wherein no resampled version of said reconstructed reference image is stored in the decoded picture buffer.
Claim 7	
7[pre]	A decoding device comprising at least one circuit configured to:
7[a]	access, from a decoded picture buffer, a reference image reconstructed at a size different from the size of a current image;
7[b]	motion compensate a reference block of said reconstructed reference image by applying a single horizontal filter GFH and a single vertical filter GFv successively on the lines and on the columns of pixels of said reference block,
7[c]	decode, for a current block of said current image, a residue block from a binary stream, and
7[d]	reconstruct the current block from said residue block and from said motion compensated reference block,
7[e]	wherein said single vertical filter GFv applied on a pixel s is such that GFv(s)=MCIFv(SCFv(s)), where MCIFv is a vertical motion compensation interpolation filter and SCFv is a vertical resampling filter, MCIFv and SCFv being applied jointly and

Reference	Limitation
7[f]	wherein said single horizontal filter GFH applied on a pixel u is such that GFH(u)=MCIFH(SCFH(u)), where MCIFH is a horizontal motion compensation interpolation filter and SCFH is a horizontal resampling filter, MCIFH and SCFH being applied jointly and
7[g]	wherein no resampled version of said reconstructed reference image is stored in the decoded picture buffer.
Claim 8	
8[pre]	A coding device comprising at least one circuit configured to:
8[a]	access, from a decoded picture buffer, a reference image reconstructed at a size different from the size of a current image;
8[b]	motion compensate a reference block of said reconstructed reference image by applying a single horizontal filter GFH and a single vertical filter GFv successively on the lines and on the columns of pixels of said reference block,
8[c]	calculate, for a current block of said current image, a residue block from said current block and from said motion compensated reference block, and
8[d]	code the residue block in a binary stream,
8[e]	wherein said single vertical filter GFv applied on a pixel s is such that GFv(s)=MCIFv(SCFv(s)), where MCIFv is a vertical motion compensation interpolation filter and SCFv is a vertical resampling filter, MCIFv and SCFv being applied jointly and
8[f]	wherein said single horizontal filter GFH applied on a pixel u is such that GFH(u)=MCIFH(SCFH(u)), where MCIFH is a horizontal motion compensation interpolation filter and SCFH is a horizontal resampling filter, MCIFH and SCFH being applied jointly and
8[g]	wherein no resampled version of said reconstructed reference image is stored in the decoded picture buffer.

When a digital video is transmitted—whether over cable lines, satellite, or the internet—extensive behind-the-scenes processing occurs to ensure the video reaches its destination efficiently. Throughout the early 2000s, when video was still primarily transmitted over dedicated cable lines, as opposed to the shared internet lines, the MPEG-4 (or counterpart H.264) video codec handled much of the world's video processing.

But after a decade in service and the proliferation of mobile devices that demanded video access on-the-go, MPEG-4 showed its age. In 2010, the industry decided to create a more efficient codec called "High Efficiency Video Codec," or "HEVC" for short.[1] The plans for HEVC were widely publicized and drew significant attention from industry groups, allowing those qualified and interested to participate in the standardization process. Even those who did not actively participate in the standardization process could follow along with the developments that were made publicly available.

U.S. Patent No. 10,250,877 ("the '877 Patent"; Ex-1001) was one of many patents that resulted from the HEVC standardization process, but it did not actually claim anything new. Not only had the technical issues identified by the inventors been solved long before for past video-coding standards, but the '877 Patent simply

---

[1] HEVC is also known as ITU-T H.265. Ex-1003, ¶59.

took what others had already proposed for inclusion in the HEVC standard and combined that with those prior solutions. The ’877 Patent’s purported addition did not make something that was old new again, and Amazon.com, Inc. and Amazon.com Services LLC (“Amazon” or “Petitioner”) therefore petitions for *inter partes* review (“IPR”) of claims 1, 4, 7, and 8 (“Challenged Claims”) of ’877 Patent.

I. GROUNDS FOR STANDING

Petitioner certifies the ’877 Patent is available for IPR, and Petitioner is not barred or estopped from requesting IPR on the grounds herein. Petitioner has not been served with a complaint by Patent Owner InterDigital, Inc. (hereafter, “PO”) asserting the ’877 Patent.

II. IDENTIFICATION OF CHALLENGE

A. Prior Art

Petitioner respectfully requests cancellation of the Challenged Claims over:

Prior Art
Nakagawa (U.S. Patent No. 5,805,222), filed on November 4, 1996, issued on September 8, 1998; prior art under at least § 102(b).[2]
Sita (U.S. Application No. 2004/0150747), filed on September 26, 2003, published on August 5, 2004; prior art under at least § 102(b).
Wiegand (Thomas Wiegand et al., *WD3: Working Draft 3 of High-Efficiency Video Coding*, JCTVC-E603 v.8), published on June 27, 2011; prior art under at least § 102(a).
Davies (Thomas Davies, *Resolution switching for coding efficiency and resilience*, JCTVC-F158 v.4), published July 7, 2011; prior art under at least § 102(a).
Fimoff (U.S. Patent No. 6,628,714), filed on December 15, 1999, issued on September 30, 2003; prior art under at least § 102(b).

B. Grounds for Challenge

Ground		Claims	Prior Art
1	§103	1, 4, 7, and 8	Nakagawa in view of Sita
2	§103	1, 4, 7, and 8	Wiegand in view of Davies and Fimoff
3	§103	1, 4, 7, and 8	Wiegand in view of Davies and Sita

---

[2] All references to pre-AIA 35 U.S.C. §§102, 103. Because the ’877 Patent is a transitional application, even if post-AIA statutes apply, that would change only which provision under § 102 each reference qualifies as prior art. *See Platinum Optics Tech. Inc. v. Viavi Sols. Inc.*, No. IPR2021-00631, 2022 WL 5056729, at *3 n.5 (PTAB Oct. 3, 2022).

III. ’877 PATENT OVERVIEW

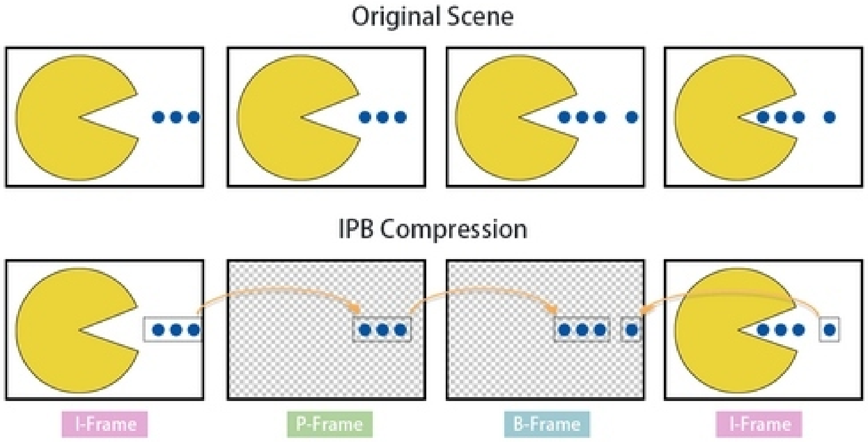
A. Technology Background

The ’877 Patent is directed to encoding and decoding videos. To store or send digital videos, the video must first be “encoded” into digital data—binary zeros and ones. Ex-1003, ¶¶32-34; *see* Ex-1001, Fig. 1 (prior-art encoder). When playing a digital video, the data is “decoded” to reconstruct the video. Ex-1003, ¶34; Ex-1001, 1:15-22, Fig. 2 (prior-art decoder).

Because videos are just a series of static images played in rapid succession, encoding each video image—also called a “picture” or “frame”— separately would require a significant amount of data. Ex-1003, ¶33. For this reason, videos are often “compressed” during encoding to reduce the amount of data needed to reconstruct them later. Ex-1003, ¶¶35-36; Ex-1001, 1:26-32.

One well-known way to compress a video is called “predictive” coding, which embodies the theory that the contents of each image in a video do not often change significantly. Ex-1003, ¶¶41-43; Ex-1020, 1-11; Ex-1021, 19-20. For instance, in a movie scene, the background may stay the same, and any objects in the scene may move slightly with each subsequent image. Encoding the *differences* between video images—such as an object’s motion—requires far less data than encoding each image anew. Ex-1003, ¶¶42-43; *see* Ex-1001, 1:50-51. An encoder characterizes the differences between images in two ways: (1) how objects move (calculated as a "motion vector"); and (2) any "residual" differences left over after accounting for an object's motion. Ex-1003, ¶¶44-45, 48-53.

With predictive coding, the encoder needs to encode at least one complete image (called an "I" frame).[3] Ex-1003, ¶¶46-47, 51. Subsequent similar images (called "P" or "B" frames) can then be encoded by using an already-encoded frame as a "reference" and calculating any motion vectors and residual differences between the reference image and the current image being encoded. *Id.* For P and B frames, these motion vectors and residual differences are sent to the decoder for reconstruction. *Id.*, ¶53.



Ex-1065, 2.

---

[3] In general, every camera cut or change of scene typically requires a new I frame, since much of the content will change. Ex-1003, ¶47.

To reconstruct a video, a decoder receives and reconstructs at least one complete I frame. *See* Ex-1003, ¶¶51-53. After that, the decoder reconstructs subsequent P or B frames using previously reconstructed frame(s) as a reference, applying the motion vectors provided by the encoder to shift the location of objects in the reference frame (called "motion compensation"), and then applying the residual to account for any other remaining differences. *Id.*; *see also* Ex-1011, 1:43-47; Ex-1014, Figs. 2, 5, ¶¶54-58; Ex-1007, 5:57-53, Ex-1033, 73. To perform predictive coding, both the encoder and decoder store reference frames in a (coded or decoded) "picture buffer"—a designated area of computer memory. Ex-1001, 1:48-49.

Another well-known way to compress video was to reduce the size (i.e., resolution) of the images through "resampling," and specifically "subsampling" or "downsampling." Ex-1003, ¶¶37-40; Ex-1019, 4-5, 12, 19-25; Ex-1064, 3. Downsampling reduces the number of pixels in each image, and therefore less data is needed for encoding. Ex-1003, ¶38. Although this may produce lower-quality videos, that is sometimes necessary. For example, if a network is too congested to transmit a full-resolution video, receiving a lower-quality video is better than receiving no video. *Id.*, ¶54. To maximize video quality when conditions permit, videos can be "adaptive[ly]" encoded at different sizes in response to network conditions. *Id.*, ¶55; Ex-1022, 1-4.

B. The ’877 Patent

The ’877 Patent was filed on July 13, 2014, as a national-stage entry of PCT/EP2013/050399, filed on January 10, 2013. The ’877 Patent claims priority to a French application (No. 12 50334) filed on January 13, 2012, which is assumed to be the priority date for purposes of this Petition only.

1. Specification

The ’877 Patent addresses an issue with combining predictive and resampling compression methods. Because videos may be adaptively resampled for transmission in response to network conditions, sometimes a reference frame may be a different size than the current frame being predicted. Ex-1001, 1:51-55. To accurately predict the current frame, the decoder must resample (either “upsample” or “downsample”) the reference frame to have the same size as the current frame being predicted. *Id.*, 1:55-59. The resampled reference frame would then be stored in the “decoded picture buffer.” *Id.*, 1:59-61.

An example of this process is shown in Figure 3, reproduced below (with annotations):

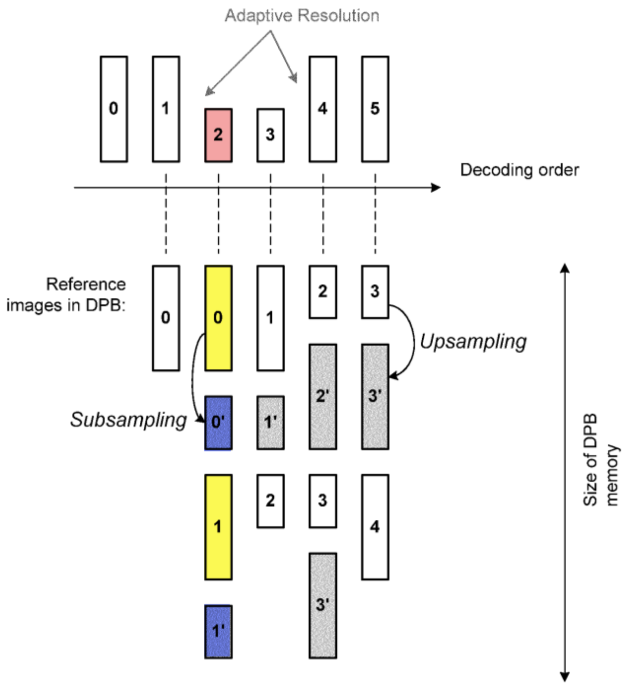


FIGURE 3

Ex-1003, ¶63. The units above the horizontal arrow represent different frames in a video stream, and the units below the horizontal arrow show the reference frames in the buffer needed to reconstruct the above-line frame. *Id.*, ¶¶64-65. As an example, reconstruction of current frame 2 (in red) uses reference frames 0 and 1 (in yellow) that are stored in the buffer. But because frames 0 and 1 are a different size than frame 2, they must first be downsampled to create smaller-sized versions labeled frames 0’ and 1’ (in blue)—also stored in the buffer—to match frame 2’s size. Frames 0’ and 1’ are then used as references to reconstruct frame 2 instead of the original-sized frames 0 and 1. *Id.*, ¶¶65-67.

According to the ’877 Patent, however, buffer memory was a scarce resource, and storing multiple versions of a reference image at different sizes would risk overloading it. Ex-1001, 2:8-13. The ’877 Patent therefore proposed combining the resampling and motion-compensation operations into a single process. *Id.*, 2:32-37. By combining these mathematical operations (which are called “filters”), reference frames could be resampled to a different size, and any motion could be compensated at the same time without needing to store the resampled reference frames in the buffer. *Id.*, 5:48-52, 6:1-7; Ex-1003, ¶¶68-69.

Moreover, because these resampling and motion-compensation “filters” are just mathematical operations that can be applied in a variety of sequences, the ’877 Patent proposed using a separate resampling/motion-compensation “filter” for each direction in the image: a vertical filter applied to each column of pixels, and a horizontal filter applied to each row of pixels. Ex-1001, 4:28-33. As the ’877 Patent recognizes, however, the particular order of the vertical and horizontal filters is immaterial. *Id.*, 4:33-36; Ex-1003, ¶70.

2. Prosecution History

The ’877 Patent originated as a national-stage entry of a PCT application. Ex-1002, 52. After a preliminary amendment, twelve claims were presented directed to decoding and encoding methods (claims 1, 6) and devices (claims 11, 13). *Id.*, 5-9. At a high level, the original claims recited, *inter alia*, modules or steps to resample and reconstruct a reference image using both a vertical and horizontal filter, in which each filter was "equal to said [] motion compensation interpolation filter composed with said [] resampling filter." *Id.*

In the first office action, the Examiner rejected all the pending claims as obvious in view of a combination of prior-art references: Ye (U.S. 8,199,812), Hong (U.S. 2013/0003847), and Karczewicz (U.S. 2009/0175336). *Id.*, 167-72. In response, the Applicant amended the claims to recite that resampling and motion compensation are performed by a *single* vertical and *single* horizontal filter. *Id.*, 189-94. The Applicant argued that not only did the prior art have the wrong configuration of filters, but none had a single vertical and single horizontal filter because they used different modules for motion compensation and resampling. *Id.*, 196-200. The Examiner was unpersuaded and issued a final rejection. *Id.*, 213-19.

The Applicant then amended the claims to specify that the reconstructed reference image must be stored in the buffer, but the resampled version must *not* be stored. *Id.*, 227-34. The Applicant also rewrote the description of the motion compensation and resampling filters into equation form. *Id.* The Applicant argued that these features sufficiently distinguished the prior art. *Id.*, 235-39. The Examiner disagreed and issued another office action using a new combination of art: Ye, Tsai (U.S. 2012/0082241), Robertson (U.S. 2010/0226437), and Sato (U.S. 6,748,018). *Id.*, 252-58.

The Applicant amended the claims again to specify that the motion compensation and resampling filters are "applied jointly" (*id.*, 274-79) and argued that the cited prior art taught only "horizontal resampling *followed by* motion compensation," *id.*, 280-82. The Examiner allowed the amended claims, concluding that "[t]he claimed invention now relates to more uniformly [sic] application jointly of either vertical and horizontal interpolation filtering and resampling to respective single filters in either vertical or horizontal applications." *Id.*, 294-97; *see also* Ex-1003, ¶¶71-78.

C. Level of Ordinary Skill in the Art

A person of ordinary skill in the art ("POSITA") would have had at least a bachelor's degree in electrical engineering, computer science, computer engineering, or a related field, and approximately three years of experience in design, development, and/or testing of relevant multimedia processing, such as video and/or image processing or coding. Ex-1003, ¶111. Advanced graduate studies with coursework and/or research focusing on topics of video encoding and decoding could substitute for work experience in the field. *Id.*

IV. CLAIM CONSTRUCTION

Claims should be construed "in accordance with the ordinary and customary meaning ... as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent." 37 C.F.R. § 42.100(b); *see also Phillips v. AWH*

Corp., 415 F.3d 1303 (Fed. Cir. 2005) (en banc). Petitioner is unaware of any prior claim construction determination related to the ’877 Patent. *See* 37 C.F.R. § 42.100(b). Nor does Petitioner believe the Challenged Claims need construction to demonstrate their unpatentability. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (explaining the Board “need only construe terms that are in controversy, and only to the extent necessary to resolve the controversy”) (cleaned up).

In separate proceedings between PO and a third party, the parties agreed that the term “circuit configured to” in the preamble of claims 7 and 8 should be given its “[p]lain and ordinary meaning (i.e., requires pre-existing programming of hardware and software to perform the cited functionality).” Ex-1012, 2. Whether the PTAB adopts this clarification of the plain meaning would have no impact on this proceeding because, as set forth below, the prior art expressly taught the claimed circuit.

V. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Obviousness over Nakagawa in view of Sita

1. Nakagawa

Nakagawa is a U.S. patent that was filed on November 4, 1996, and issued on September 8, 1998. Nakagawa therefore constitutes prior art under at least pre-AIA §102(b).

Nakagawa explains that when a video contains sudden or significant content changes, encoding the change may require "an overwhelming amount of coded frame data that exceeds a standard level allowed for each frame," thereby causing "extreme degradation of image quality." Nakagawa, 1:20-30; Ex-1003, ¶81. According to Nakagawa, one way to avoid this problem is to alter the size of the encoded images in real time to "regulate the amount of coded data at an appropriate level." Nakagawa, 2:49-53; Ex-1003, ¶82.

To accomplish this, Nakagawa describes "a video coding apparatus for performing predictive coding" that "selects picture resolution to be used in coding of a source picture of a current frame." Nakagawa, Abstract; Ex-1003, ¶83. Specifically, Nakagawa explains how—using the existing H.261 video-coding standard as an example—video images can be encoded to be of one of two sizes ("CIF" or "QCIF"). Nakagawa, 4:43-51; Ex-1003, ¶83. After encoding each frame at a particular size, Nakagawa's system evaluates a set of factors to determine whether to use a different size for the next frame to be encoded. *See, e.g.*, Nakagawa, 3:64-4:6; Ex-1003, ¶83. Because Nakagawa's system may switch the size of each encoded frame, it will sometimes predict a current frame using a different-sized reference frame. Nakagawa, 6:15-40; Ex-1003, ¶84. Nakagawa therefore provides a "CIF/QCIF converter" that "retrieves the reconstructed picture of the previous frame from the frame memory 22 and converts it" to match the size of the frame being predicted (Nakagawa, 6:15-40) before performing conventional reconstruction operations (such as calculating motion vectors and the residual) to encode the current frame. *See id.*, 4:52-5:20; Ex-1003, ¶84.

2. Sita

Sita is a U.S. patent application publication that was filed on September 26, 2003, and published on August 5, 2004. Sita therefore constitutes prior art under at least pre-AIA §102(b).

Sita discloses a decoder that can receive many different types of video formats (including high-definition television signals) and convert them into a lower-resolution, standard-definition signal. *See* Sita, ¶¶2-3, 66-67; Ex-1003, ¶¶87-88. In other words, an encoder may send high-definition video, which Sita's decoder reconstructs and converts to a standard-definition format.

To achieve this, Sita discloses a downsampling operation to reduce the size of the video image after it has been decoded and reconstructed. *See* Sita, ¶¶67, 82; Ex-1003, ¶88. These downsampled images are not only output by the decoder (for compatibility with standard-definition displays), but they are also stored in memory to serve as reference frames to predict subsequent images. Sita, ¶¶82-83; Ex-1003, ¶88. Because Sita's decoder may receive a signal from the encoder representing high-definition images, yet all reference images are stored at standard definition, Sita also discloses an image-reconstruction process to handle the mismatch between the two. Ex-1003, ¶89. In particular, Sita explains how a combined resampling/motion-compensation operation is performed to convert the standard-definition reference images stored in memory back into high definition so that the high-definition resolution motion vectors from the encoder can be applied to reconstruct a prediction of the current image. Sita, ¶¶88, 89, 110-112, Fig. 2B; Ex-1003, ¶89. Sita further discloses that this process can be applied with two separate, one-dimensional filters: one in the horizontal direction, and one in the vertical direction. *See, e.g.*, Sita, ¶95; Ex-1003, ¶90.

3. Claim 1

a. Element 1[pre]

To the extent limiting, Nakagawa discloses or suggests element 1[pre].

For the "decoding method," Nakagawa discloses a "video coding apparatus that performs predictive coding." Nakagawa, 1:4-8.

Although Nakagawa describes an *encoder* in detail, a POSITA would have understood that Nakagawa describes a decoding method within the encoder, and thus also implicitly describes a decoder and the process by which the decoder would decode a bitstream into video. Ex-1003, ¶¶117-24. Specifically, in Figure 2 (reproduced with annotations), Nakagawa provides a diagram of the encoder that has a built-in decoding loop.[4] *Id.*, ¶¶122. But whereas an encoder would typically output a bitstream—represented as the "to transmission line" at 18 in Figure 2—Nakagawa also shows a *separate* output from the decoding loop—represented as the "to display unit" at 26. *See* Nakagawa, Fig. 2; Ex-1003, ¶121. A POSITA would have therefore understood that Nakagawa is showing both the encoding *and* decoding processes in Figure 2. Ex-1003, ¶124. Moreover, a POSITA would have readily understood which components would be needed at the decoder, identified in red below:

---

[4] Encoders in predictive-coding systems include a decoding loop to generate *reconstructed versions* of video images—which may differ slightly from the original versions—to calculate motion vectors and residuals, because those vectors and residuals will be applied to reconstructed versions of the image at the decoder. Ex-1003, ¶¶49-52. The images the encoder and decoder work from need to be the same. *Id.*, ¶49.

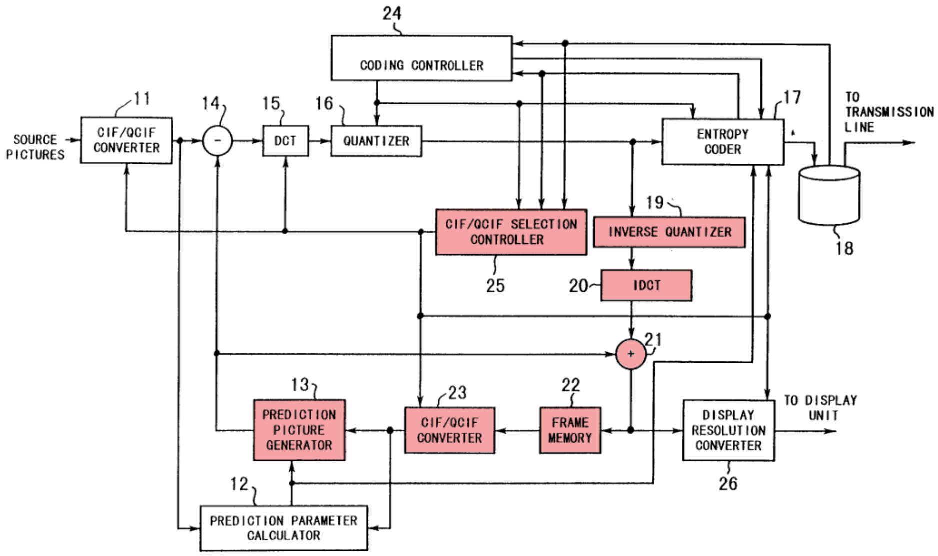


FIG. 2

Id., ¶122; *see also* Nakagawa, 5:42-59, 6:15-40 (describing the decoding process in the context of the encoder). A POSITA therefore would have understood how (and thus found it obvious) to implement a decoder that would operate in accordance with Nakagawa because Nakagawa effectively described the decoder built into the encoder. Ex-1003, ¶124.

Nakagawa further discloses that "the coding system generates a digital video bit stream" that is sent to the decoder for reconstruction. Nakagawa, 3:60-63; *see also id.*, 5:38-41. By disclosing a "bit stream," a POSITA would have understood that Nakagawa discloses reconstructing video images at the decoder from a "binary stream" as claimed, because a POSITA would have understood that "bits" are binary values. Ex-1003, ¶125.

Nakagawa further discloses "reconstruct[ing] a current block of a current image from a reference block of a reference image reconstructed at a different size from the size of said current image," as claimed. Specifically, Nakagawa discloses that, as part of the encoding process, "a source picture is divided into blocks of pixels," Nakagawa, 1:9-20, and involves "[c]omparing the source picture of the current frame with the reconstructed picture of the previous frame on a block-by-block basis," *id.*, 4:62-66; *see also id.*, 4:43-51. Nakagawa then discloses that, within the decoding loop, images are reconstructed based on a reference image. *See id.*, 5:2-12 (noting motion vectors applied to reference image to produce a "prediction picture"), 5:42-59 (noting how a "reconstructed picture generator 21 reconstructs a picture by adding the prediction picture…and the [residual]"). By disclosing a block-based reconstruction process as part of the decoding loop, a POSITA would have understood that a decoder performs this reconstruction operation as well. Ex-1003, ¶¶126-27.

Nakagawa also discloses that the current and reference frames can be of at least one of two sizes (CIF or QCIF) (Nakagawa, 4:43-51) and that the resolution can be switched mid-video stream. *See, e.g.*, *id.*, 7:52-55. Consequently, a block for a current image at one resolution (e.g., CIF) will sometimes be reconstructed from a block using a reference image at a different resolution (e.g., QCIF), and Nakagawa explains how to resample the reference image to match the resolution of the current frame so that the current frame can be reconstructed. *See, e.g.*, *id.*, 6:15-40. A POSITA would have understood that if the current and reference pictures have different sizes at the encoder, they would also have different sizes when being reconstructed at the decoder. Ex-1003, ¶¶128-29.

Finally, Nakagawa discloses storing reconstructed reference images in a picture buffer, which Nakagawa calls a "buffer" or "frame memory." *See, e.g.*, Nakagawa, 4:58-59 ("A reconstructed (or decoded) picture of the previous frame stored in a frame memory 22"); *see also id.*, 2:55-62, 4:58-61; Ex-1003, ¶130. A POSITA would have understood that a decoder would have a buffer or frame memory for storing reference pictures as well. Ex-1003, ¶131.

b. Element 1[a]

The combination of Nakagawa and Sita renders element 1[a] obvious.

Nakagawa discloses calculating motion vectors at the encoder by "[c]omparing the source picture of the current frame with the reconstructed picture of the previous frame on a block-by-block basis." Nakagawa, 4:62-65; Ex-1003, ¶¶132-33. Nakagawa then discloses that, as part of the decoder loop, a "prediction picture generator" receives the motion vectors and "produces a prediction picture of the current frame" by applying the calculated motion vectors. Nakagawa, 5:4-12; Ex-1003, ¶133.

Because it is possible that, during the motion-compensation process, the reference image may have a different resolution (i.e., size) than the current image being predicted, Nakagawa discloses a resolution converter ("CIF/QCIF converter") that changes the size of the reference image to match that of the current image. *See* Nakagawa, 6:15-40 (resampling operation to ensure the current and reference blocks are the same size), 5:4-12 (describing application of motion vectors), Fig. 2 (showing a CIF/QCIF converter (23) for resampling, and prediction picture generator (13) for motion vectors); Ex-1003, ¶134.

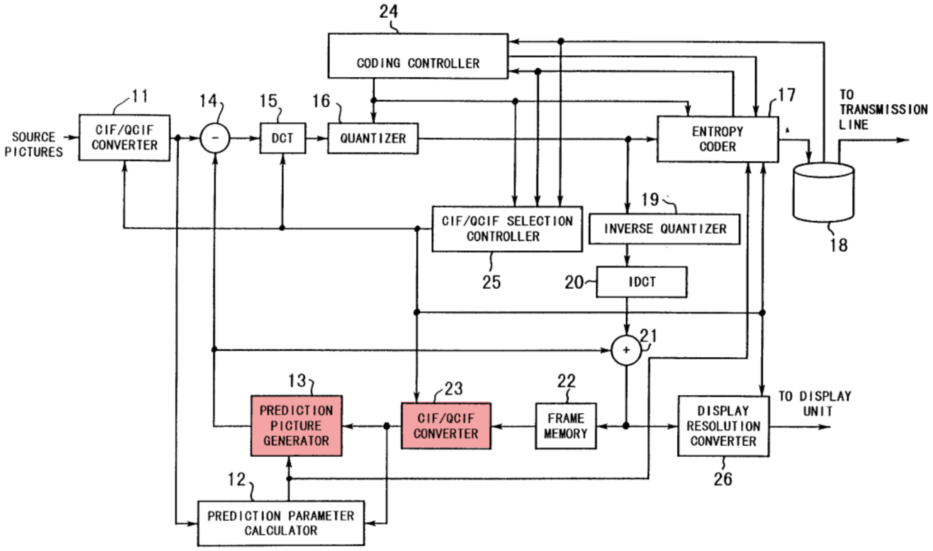
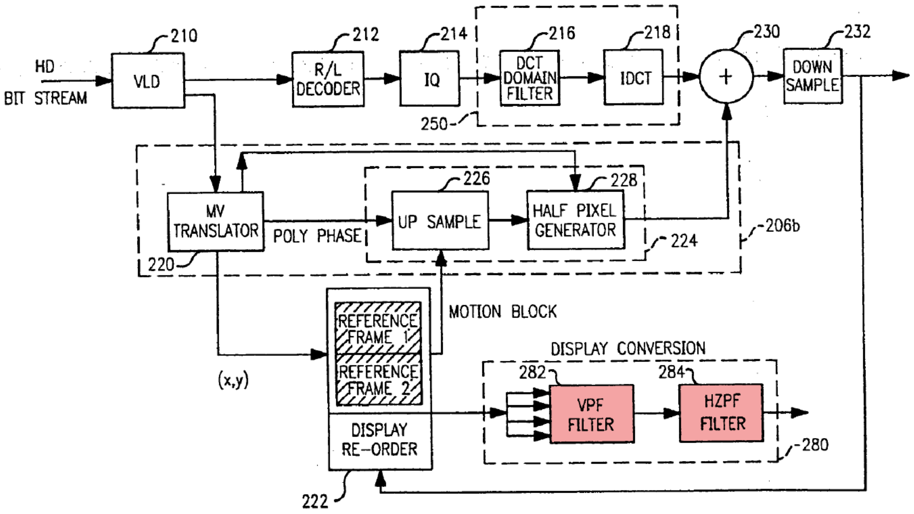


FIG. 2

However, Nakagawa does not expressly disclose the details of implementing the resolution conversion during the motion-compensation process. Ex-1003, ¶135.

A POSITA would have understood that changing the resolution of a reference frame to match that of the current frame during a motion-compensation process was known in the prior art, and others had expressly provided solutions. *Id.* Indeed, when television networks began broadcasting high-definition television signals, those in the art provided processes to convert high-definition signals to standard definition images to support older televisions, and use standard-definition images as references to decode incoming high-definition signals to save memory. *Id.* For instance, Sita teaches a video decoder that generally operated in accordance with predictive video-coding standards, like Nakagawa. *See* Sita, ¶2; Nakagawa, 1:10-13; Ex-1003, ¶136. As part of the decoder, Sita teaches using a resampling (226) and motion-compensation-interpolation (228) filter operation (collectively, filter 224), in which a standard-definition reference block is resampled for motion compensation by applying the filter. Sita, ¶¶82, 88-89; Ex-1003, ¶136.

Although Sita does not expressly show that its filter (224) applies separate horizontal and vertical filters, as claimed (*see* Sita, Fig. 2B), a POSITA would have understood—or at least found it obvious—that Sita uses such filters. Ex-1003, ¶137. Specifically, a POSITA would have understood that because Sita identifies different image sizes with different numbers of pixels in both the horizontal and vertical direction (just like Nakagawa), Sita's resampling/motion-compensation-interpolation operation would be performed in both the horizontal and vertical directions. *See, e.g.*, Sita, ¶¶66-67 (Tables 1 and 2) (showing in Table 1 different formats with different pixel amounts in both the horizontal and vertical direction, and showing in Table 2 conversion between formats, e.g., 1125p (with 1920-by-1080 pixels) to 525p (with 640-by-480 pixels)); Ex-1003, ¶138. Sita also expressly states that a POSITA would know how to perform these operations in both directions using separate horizontal and vertical filters. Sita, ¶95; Ex-1003, ¶139. And Sita shows that, in a separate filtering process to convert the final reconstructed images to match a *display* resolution, separate vertical and horizontal resampling filters are used. Sita, Fig. 2B, ¶81; Ex-1003, ¶140. An annotated version of Sita's Figure 2B is provided below, which identifies those filters:



A POSITA would have therefore understood that Sita's resampling/motion-compensation-interpolation filter (224) would do the same, or at least would have found it obvious to configure Sita's filter (224) accordingly. Ex-1003, ¶141.

Furthermore, although Sita shows applying a vertical filter before a horizontal filter (for the separate display filter 280), a POSITA would have appreciated that the order of the horizontal and vertical filters is of little consequence. *Id.*, ¶142. Not only were only two options available—(1) horizontal followed by vertical filtering; or (2) vertical followed by horizontal filtering—but a POSITA would have understood that the order is immaterial (*id.*), as the ’877 Patent itself admits (Ex-1001, 4:33-36). The selection of a specific filtering sequence—all of which would have produced similar results—would have been well within the implementation expertise of a POSITA. Ex-1003, ¶142.

A POSITA would therefore have found it obvious to use Sita’s upsampling/motion-compensation-interpolation filtering scheme to convert, e.g., QCIF images to CIF images (or vice versa) and apply the motion vectors in a decoder consistent with Nakagawa’s disclosure. *Id.*, ¶143. And although Nakagawa did not explicitly disclose performing upsampling and motion-compensation-interpolation operations using a horizontal filter and a vertical filter, a POSITA would have found this obvious in view of Sita, which solved the exact problem of performing upsampling during motion compensation to match resolution (size) of a reference block to the current block. *Id.*

But a POSITA would have also found it obvious to combine Nakagawa and Sita for another reason: Sita would have made it easier for Nakagawa to implement other known coding standards, such as MPEG-1 or MPEG-2. Nakagawa describes the embodiments of its invention by “adopt[ing] a coding scheme of the ITU-T standard H.261." Nakagawa, 4:43-44. A POSITA would have understood, however, that Nakagawa's disclosure was not limited to H.261 but instead provided H.261 as an example; none of the video coding blocks in Nakagawa's embodiments are limited to an H.261 coding scheme. Ex-1003, ¶¶144-47; *see* Nakagawa, 1:10-13 (noting "H.261 and…MPEG-1 and MPEG-2, for example, are well-acknowledged international standards for motion picture coding techniques"), 10:3-4 ("The foregoing [disclosure] is considered as illustrative only of the principles of the present invention.").

A POSITA would have also understood that newer video coding standards (e.g., MPEG-1 or MPEG-2) provided other benefits, such as better-quality video, more options for video resolutions (other than just CIF/QCIF for H.261), more options to implement those coding standards due to their wider adoption, and more-refined motion-compensation operations than H.261 (because MPEG-1 and MPEG-2 used half-pixel motion compensation, rather than full-pixel compensation like H.261). Ex-1003, ¶145. A POSITA therefore would have found it obvious to configure Nakagawa's system to implement these other coding schemes that Nakagawa expressly identified. *Id.*, ¶147. A POSITA would have appreciated that Sita's filtering operations were already compliant with the MPEG-1 and MPEG-2 coding standards, and specifically already implemented half-pixel motion-compensation-interpolation operations used by these standards. *See* Sita, ¶2 (describing an MPEG-2 compliant scheme); Ex-1003, ¶148.

A POSITA would have had a reasonable expectation of success in incorporating Sita's filtering operation into Nakagawa's decoding loop (and, by extension, a compatible decoder) because Nakagawa already performed the same resampling/motion-compensation operations as Sita, just as two separate steps. Sita explained how to combine them into a single step and apply them one direction at a time. The modification would have involved little more than computer coding to operate on simple data structures, such as arrays, which would have well been within the skill of a POSITA. Ex-1003, ¶149.

c. Element 1[b]

Nakagawa discloses or suggests element 1[b].

Nakagawa discloses an encoder that generates a residue block, which Nakagawa calls a "prediction error signal." Nakagawa, 5:13-17 ("The prediction error calculator 14 produces a prediction error signal by calculating differences between the current frame picture…and the prediction picture"); Ex-1003, ¶¶150-51. Indeed, Nakagawa's "prediction error signal" is calculated the same way as the "residue block" in the '877 Patent. *See* Ex-1001, 9:5-11 ("the residue block is obtained by subtracting the prediction block pixel by pixel from the current block"); Ex-1003, ¶152.

Nakagawa also discloses that, as part of the decoder loop, a bitstream of "quantized transform coefficients" are decoded "to reproduce the prediction error signal." Nakagawa, 5:42-47. A POSITA would have also understood that implementing a decoder consistent with Nakagawa would have similarly decoded a "prediction error signal" because that is the conventional process to reconstruct a current block based on a predicted reference block. Ex-1003, ¶153; Nakagawa, 5:47-53; Sita, ¶78.

As a general matter, though, the use of residue blocks in image decoding and reconstruction was well-known, and the '877 Patent did not improve or modify the use or management of residue blocks in the decoding process. Ex-1003, ¶¶152, 154.

d. Element 1[c]

Nakagawa discloses or suggests element 1[c].

As explained in Sections V.A.3.a (1[pre]) and V.A.3.b (1[a]), Nakagawa discloses a decoding loop in which a "prediction picture generator" performs motion compensation to produce a "prediction picture" comprising motion-compensated reference blocks. *See also* Ex-1003, ¶¶155-57. And as explained in Section V.A.3.c (1[b]), Nakagawa also discloses that, as part of the encoder's decoding loop, "[a] reconstructed picture generator 21 reconstructs a picture by adding the prediction picture produced by the prediction picture generator 13 and the prediction error signal," Nakagawa, 5:47-53, otherwise known as a "residue" or "residual." *See also* Ex-1003, ¶156.

Reconstructing an image by combining a motion-compensated reference block and a residual block was conventional and well-known before the '877 Patent. Ex-1003, ¶158. And Nakagawa's reconstruction process (in the encoder's decoding loop) is consistent with Sita's reconstruction process in a decoder, which reconstructs the current block from the residual block and motion-compensated reference block. Sita, ¶78 (explaining how "Motion Compensation Processor 206 receives [a prediction block] and then adds [the residual block]…to produce…[a block for] the current video image[.]"); Ex-1003, ¶158.

Accordingly, a POSITA would have understood, or at least found it obvious, that upon implementing a decoder consistent with Nakagawa, a current block (Nakagawa's "reconstructed picture") would be reconstructed by combining a residue block (Nakagawa's "prediction error signal") and motion-compensated reference block (Nakagawa's "prediction picture"), as claimed. Ex-1003, ¶159.

e. Element 1[d]

The combination of Nakagawa and Sita renders element 1[d] obvious.

As explained in Section V.A.3.b (1[a]), combining Nakagawa and Sita renders obvious "applying…a single vertical filter GFv successively…on the columns of pixels of said reference block," as claimed. *See also* Ex-1003, ¶¶160-62.

Although written using equations, element 1[d] effectively recites that the single vertical filter in element 1[a] comprises a vertical motion-compensation-interpolation filter (MCIFv) and a vertical resampling filter (SCFv) combined into one. Ex-1003, ¶161; *see also* Ex-1001, 5:57-67 ("[T]he horizontal filters $MCIF_H$ and SCF_H are grouped into a single filter GF_H" using, e.g., equation $GF_H = MCIF_H \, o \, SCF_H$ and the same for vertical filter "V," "where (f *o* h)(x) = f(h(x))."); *see also* Ex-1067, 525:11-527:21 (PO's expert explaining the operation of the joint filter).

Nakagawa teaches that, as part of the decoding loop, a vertical resampling operation is performed. *See, e.g.*, Nakagawa, 4:46-51 (noting how CIF pictures have a vertical resolution of 288 pixels, and QCIF pictures have a vertical resolution of 144 pixels), 6:15-40 (describing a resampling operation to convert CIF to QCIF, and vice-versa); Ex-1003, ¶163. After resampling, a motion-compensation operation is performed. Nakagawa, Fig. 2 (showing CIF/QCIF converter 23 followed by prediction-picture generator 13), 5:2-12; Ex-1003, ¶163. However, Nakagawa does not explain how the CIF/QCIF converter and prediction-picture generator operate, nor does Nakagawa expressly describe motion-compensation *interpolation* (because Nakagawa uses full-pixel motion vectors, which do not require interpolation). Ex-1003, ¶164.

Sita, however, provides those details for a decoder that, like Nakagawa, implements reference-frame upsampling for a motion-compensation procedure, including motion-compensation interpolation to apply half-pixel motion vectors. Ex-1003, ¶164. Specifically, Sita provides examples of encoding a current frame using a different-sized reference frame scaled by a particular ratio. *See* Sita, ¶109 (describing 3:1 or 2:1 upsampling of a reference image to match the size of the current image); Ex-1003, ¶164. Sita then provides filter weights in Tables 7B (for 3:1 upsampling) and 7C (for 2:1 upsampling) that, taking only the reference-frame pixel values as inputs, both upsamples and performs motion-compensation interpolation as part of a single operation, and in a single computer-processor clock cycle. Sita, ¶¶110-12; Ex-1003, ¶¶165-81; Ex-1067, 525:11-527:21 (PO's expert explaining the joint filter applies a set of weights to pixel values in a reference picture, just like Sita). And although Sita expressly describes these operations in the context of a horizontal filter, Sita recognizes the same teachings apply to a vertical filter. Sita, ¶95; Ex-1003, ¶181.

A POSITA would have been motivated to modify Nakagawa's decoding loop (and, accordingly, also implement at a decoder) to perform Sita's joint resampling/motion-compensation-interpolation filter operations. Ex-1003, ¶182. Specifically, a POSITA would have understood that Sita provides a compact method in which pixels from a different-sized reference frame are retrieved for a combined upsampling and half-pixel interpolation operation. *Id.*; *see* Sita, ¶108; *see also* §V.A.3.b (1[a]) (explaining the benefits of modifying Nakagawa to implement MPEG-2, including its half-pixel motion vectors). A POSITA would have recognized that Sita's joint operations would result in reduced use of computational resources because a lower memory bandwidth would be needed for fetching reference-frame pixels while at the same time streamlining the implementation of the resampling/motion-compensation-interpolation process. Ex-1003, ¶182. Furthermore, by combining these operations, multiple pixels could be operated upon in a single clock cycle, which would have supported Nakagawa's express goal of reducing the latency of these operations. *See, e.g.*, Sita, ¶110 (noting a single clock cycle can output sub-pixel values at 6 (or 4) locations); Nakagawa, 1:32-2:46 (explaining a problem with the prior art is the delay in deciding to adapt the video-image resolution during encoding, and that "prior-art system must execute each process faster than normal pipelined systems in order to finish the frame coding within a fixed cycle time"); Ex-1003, ¶182.

A POSITA would have had a reasonable expectation of success in implementing Sita's joint operations in Nakagawa's coding system because it merely involves combining two mathematical operations into one by selecting appropriate weights for pixel values to produce the same output as separate resampling and motion-compensation interpolation equations would have. Ex-1003, ¶183. And Sita provides two sets of exemplary weights for certain resampling ratios to choose from, if desired. *Id.* In other words, a POSITA would need only modify some mathematical equations in computer code, which would have been well within the skill of a POSITA. *Id.*

f. Element 1[e]

The combination of Nakagawa and Sita renders element 1[e] obvious.

As discussed previously, the combination of Nakagawa and Sita renders obvious element 1[d] (*see* §V.A.3.e (1[d])), and 1[e] differs from 1[d] only with respect to the direction the filter is being applied. Ex-1003, ¶¶184-185. Furthermore, as also explained previously, a POSITA would have understood, or at least found it obvious, to apply Sita's filtering operations as separate horizontal and vertical filters. *See* §V.A.3.b (1[a]). But whereas a POSITA would need to configure Sita's express horizontal filter to apply in the vertical direction (*see* §V.A.3.e), Sita expressly describes a joint resampling/motion-compensation-interpolation operation in the horizontal direction. Ex-1003, ¶186.

g. Element 1[f]

The combination of Nakagawa and Sita renders element 1[f] obvious.

As explained in Section V.A.3.a (1[pre]), Nakagawa discloses that reconstructed reference images are stored in "frame memory," and that when a reference image in frame memory is a different size than the current frame, the reference image is resampled to match the size of the current frame. *See also* Nakagawa, 6:15-40; Ex-1003, ¶¶187-88.

As Figure 2 of Nakagawa further shows, because "frame memory" provides the input to an external "CIF/QCIF converter," and the converter does not provide an input back to the "frame memory" while converting reference frames only as needed (Nakagawa, 6:15-34), a POSITA would have understood that no resampled version of the reference image is stored back in memory. Ex-1003, ¶189. An annotated version of Figure 2 is provided below, which shows the signal flow from the frame memory to the CIF/QCIF converter:

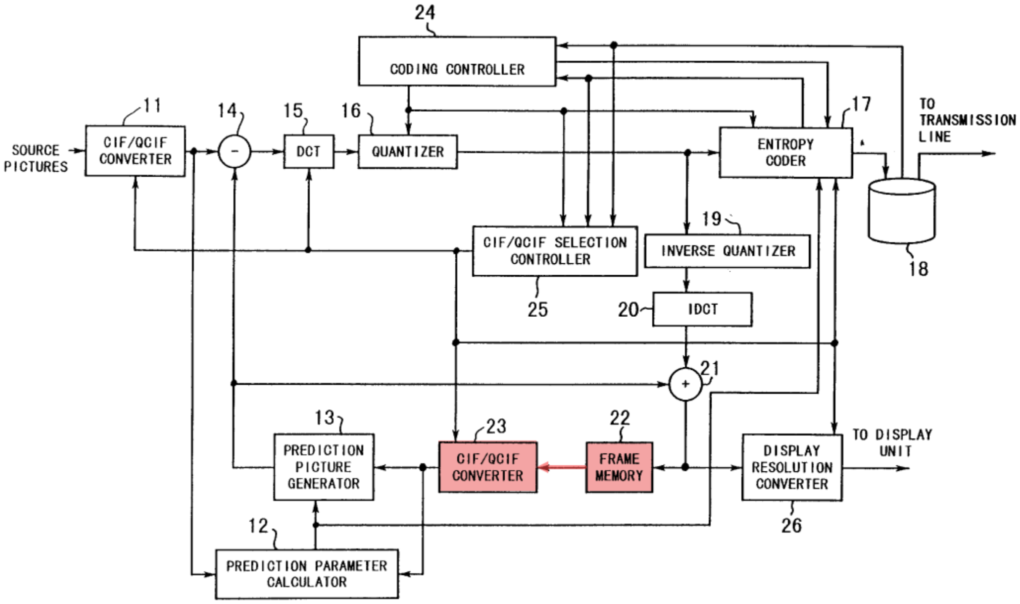
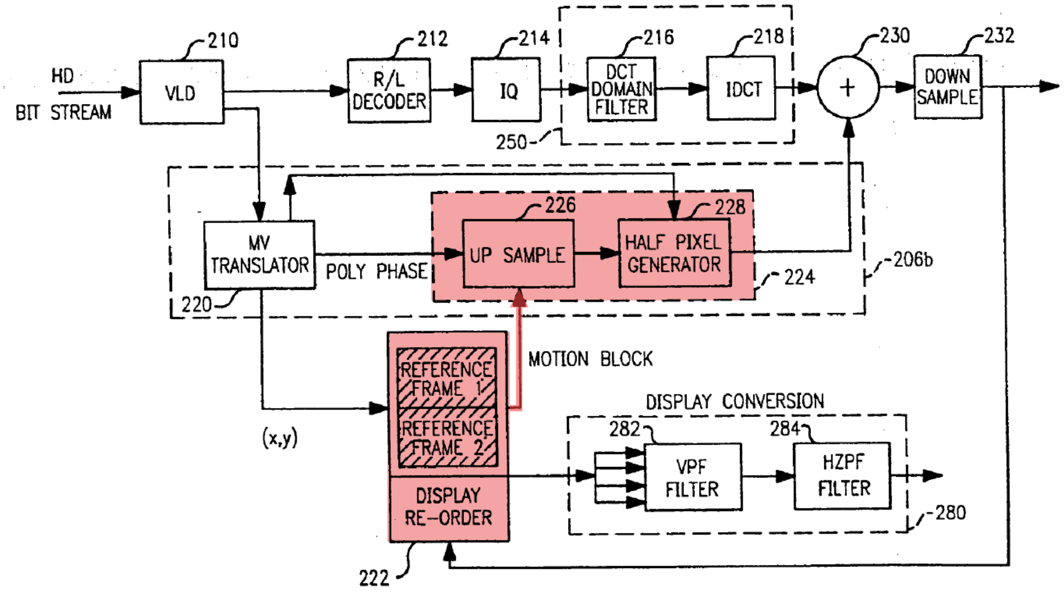


FIG. 2

Moreover, a POSITA would have appreciated that, upon modifying Nakagawa to implement Sita's joint resampling/motion-compensation-interpolation filter, there would be no need to store a resampled version of the reference image in memory. Ex-1003, ¶190. Like Nakagawa, Sita's reference frame memory provides a reference frame as an input to the joint filter (224) without any feedback to the memory. An annotated version of Sita Figure 2B is provided below:



A POSITA would have also understood that Sita does not directly produce a "resampled version" of a reference image, as claimed, because Sita performs both resampling and motion-compensation-interpolation operations together. Ex-1003, ¶190.

Finally, as discussed previously in Section V.A.3.a (1[pre]), although Nakagawa primarily describes an encoder, a POSITA would have understood that the same arrangement would be implemented at the decoder because Nakagawa describes this arrangement in the context of a decoding loop. Ex-1003, ¶191.

4. Claim 4

The combination of Nakagawa and Sita renders this claim obvious as a whole.

Although the analysis of claim 1 focused on applying Nakagawa to a decoder, as discussed above, Nakagawa primarily describes an encoder, and therefore expressly teaches the relevant claim elements as part of an encoder. *See* §V.A.3.a (1[pre]); Ex-1003, ¶¶192-93.

a. Element 4[pre]

To the extent limiting, Nakagawa discloses element 4[pre]. *See* §§V.A.3.a (1[pre]), V.A.4 (claim 4, generally); Ex-1003, ¶194.

b. Element 4[a]

The combination of Nakagawa and Sita renders element 4[a] obvious. *See* §V.A.3.b (1[a]); Ex-1003, ¶195.

c. Element 4[b]

Nakagawa discloses element 4[b].

Nakagawa discloses that "[t]he prediction error calculator 14 produces a prediction error signal by calculating differences between the current frame picture provided by the CIF/QCIF converter 11 and the prediction picture provided by the prediction picture generator 13." Nakagawa, 5:13-17; *see also id.*, 5:2-12 (explaining how prediction picture generator applies motion vectors to a reference frame to produce the "prediction picture of the current frame"). As stated previously, a

POSITA would have understood that Nakagawa's "prediction error signal" is a "residue block," and the "prediction picture" is a "motion compensated reference block," as claimed. *See* §V.A.3.c (1[b]); Ex-1003, ¶¶196-98.

d. Element 4[c]

Nakagawa discloses element 4[c].

Nakagawa discloses that the "prediction error signal" is sent through a "DCT processor" that produces "transform coefficients," which are then quantized and sent to an "entropy coder" that "assigns different codes to the combinations of those received data…which will be sent out to the transmission line in the form of bit stream at a constant transfer rate." Nakagawa, 5:17-41. A POSITA would have understood this process describes the coding of a residue block into a binary stream, as claimed. Ex-1003, ¶¶199-200.

e. Element 4[d]

The combination of Nakagawa and Sita renders element 4[d] obvious. *See* §V.A.3.e (1[d]); Ex-1003, ¶201.

f. Element 4[e]

The combination of Nakagawa and Sita renders element 4[e] obvious. *See* §V.A.3.f (1[e]); Ex-1003, ¶202.

g. Element 4[f]

Nakagawa discloses element 4[f]. *See* §V.A.3.g (1[f]); Ex-1003, ¶203.

5. Claim 7

a. Element 7[pre]

To the extent limiting, Nakagawa discloses, or at least renders obvious, element 7[pre].

Nakagawa discloses a "video coding apparatus for performing predictive coding of digital video input signals." Nakagawa, Abstract. Because Nakagawa discloses processing digital signals, a POSITA would have understood, or at least found it obvious, that Nakagawa discloses a system implemented on specialized hardware or computer processors and memory—which alone or together form a circuit—operating pre-programmed software. Ex-1003, ¶¶204-06; *see* §IV (Claim Construction). Moreover, as discussed previously, a POSITA would have found it obvious to implement the disclosures of Nakagawa as part of a decoding device, in addition to an encoding device. *See* §V.A.3.a (1[pre]); Ex-1003, ¶207.

b. Element 7[a]

Nakagawa discloses, or at least renders obvious, element 7[a]. *See* §V.A.3.a (1[pre]) (discussing use of a decoded picture buffer for reference frames reconstructed at a size different than the current frame, and the implicit and/or obvious implementation at a decoder). Nakagawa also discloses accessing those reference images from a buffer or frame memory (e.g., a decoded picture buffer, as claimed) when reconstructing an image. *See, e.g.*, Nakagawa, 5:4-7 ("The prediction picture generator 13 receives…a decoded image of the previous frame from the frame memory 22"); Ex-1003, ¶208.

c. Element 7[b]

The combination of Nakagawa and Sita renders element 7[b] obvious. *See* §V.A.3.b (1[a]); Ex-1003, ¶209.

d. Element 7[c]

Nakagawa discloses, or at least renders obvious, element 7[c]. *See* §V.A.3.c (1[b]); Ex-1003, ¶210.

e. Element 7[d]

Nakagawa discloses, or at least renders obvious, element 7[d]. *See* §V.A.3.d (1[c]); Ex-1003, ¶211.

f. Element 7[e]

The combination of Nakagawa and Sita renders element 7[e] obvious. *See* §V.A.3.e (1[d]); Ex-1003, ¶212.

g. Element 7[f]

The combination of Nakagawa and Sita renders element 7[f] obvious. *See* §V.A.3.f (1[e]); Ex-1003, ¶213.

h. Element 7[g]

Nakagawa discloses, or at least renders obvious, element 7[g]. *See* §V.A.3.g (1[f]); Ex-1003, ¶214.

6. Claim 8

a. Element 8[pre]

To the extent limiting, Nakagawa discloses, or at least renders obvious, element 8[pre]. *See* §§V.A.3.a (1[pre]) (describing details of the coding device), V.A.4 (claim 4, generally) (noting express disclosure of a coding device), V.A.5.a (7[pre]) (describing disclosure and/or obviousness of a circuit and/or preprogrammed software); Ex-1003, ¶215.

b. Element 8[a]

Nakagawa discloses element 8[a]. *See* §§V.A.3.a (1[pre]), V.A.5.b (7[a]); Ex-1003, ¶216.

c. Element 8[b]

The combination of Nakagawa and Sita renders element 8[b] obvious. *See* §V.A.3.b (1[a]); Ex-1003, ¶217.

d. Element 8[c]

Nakagawa discloses element 8[b]. *See* §V.A.4.c (4[b]); Ex-1003, ¶218.

e. Element 8[d]

Nakagawa discloses element 8[c]. *See* §V.A.4.d (4[c]); Ex-1003, ¶219.

f. Element 8[e]

The combination of Nakagawa and Sita renders element 8[e] obvious. *See* §V.A.3.e (1[d]); Ex-1003, ¶220.

g. Element 8[f]

The combination of Nakagawa and Sita renders element 8[f] obvious. *See* §V.A.3.f (1[e]); Ex-1003, ¶221.

h. Element 8[g]

Nakagawa discloses element 8[g]. *See* §V.A.3.g (1[f]); Ex-1003, ¶222.

B. Ground 2: Obviousness over Wiegand in view of Davies, and Fimoff

1. Wiegand

Wiegand is a working-draft proposal for the High-Efficiency Video Coding standard that was published on June 27, 2011. Ex-1045. Wiegand therefore constitutes prior art under at least pre-AIA § 102(a).[5]

Wiegand describes a new draft "industry standard for compressed video representation with substantially increased coding efficiency and enhanced

---

[5] Wiegand was publicly accessible. Ex-1005, ¶¶43-69; *see also generally* Ex-1005. Moreover, Wiegand had been cited by others around the time of its publication (Ex-1042, Ex-1047, Ex-1048, Ex-1049), demonstrating that those interested had access and knew where to find it.

robustness to network environments." Wiegand, 12 (Forward), 13 (§0.1). As part of the draft standard, Wiegand discloses a detailed decoding process. *See generally*, *id.*, 84-154 (§8).

Specifically, Wiegand discloses that the decoder first receives a bitstream (or "NAL units") from the encoder. *Id.*, 85 (§8.1). One of the first steps is to reconstruct pictures and mark them as reference pictures. *Id.* (§8.2.2). When a current frame is to be reconstructed based on a reference frame (called "inter prediction" decoding), it will first apply motion vectors to predict the current frame. *Id.*, 109 (step 1, referencing §8.4.1), 110-11 (§8.4.1 instructing to follow §8.4.2), 111-30 (describing derivation and interpolation of motion vectors). Then the decoder will decode and apply a residual signal that accounts for any remaining differences not captured by the prediction. *Id.*, 109 (step 2, referencing §8.4.3), 130-33 (§8.4.3). The picture is then reconstructed using the prediction for the current frame and the residual. *Id.*, 109 (step 4).

2. Davies

Davies is a proposal published by another HEVC-standardization participant on July 7, 2011. Davies therefore constitutes prior art under at least pre-AIA § 102(a).[6]

Davies expressly proposes modifying Wiegand—*see* Davies, 4 (§ 4) ("Various modifications to the current Working Draft (WD) [3] are needed"), 11 (Wiegand as "[3]")—to permit "resolution switching" "to allow video communications to use re-scaling to adapt seamlessly to adverse network conditions." *Id.*, 1. Specifically, Davies proposes "that an encoder can signal a change of resolution and predict frames across resolutions," in which "reference frames are simply re-scaled as needed." *Id.*, 2 (§1).

Davies recognizes that although resolution switching was permitted in past standards, past implementations required sending a special "IDR" frame to reset the reference buffer, which was "difficult to transmit in a low-delay environment," Davies, 2 (§1). Davies instead proposes using "separable down- and up-scaling

---

[6] Davies was publicly accessible. Ex-1005, ¶¶43-69; *see also generally* Ex-1005. Moreover, Davies had been cited by others around the time of its publication, Ex-1048, Ex-1047, demonstrating those interested had access and knew where to find it.

filters" to resize reference images (Davies, 4 (§3.2)), which can be applied during Wiegand's inter-prediction process. *Id.*, 7 (§4.4.1.1) (proposing changes to Wiegand § 8.4.2.1.6).

3. Fimoff

Fimoff is a U.S. patent that was filed on December 15, 1999, and issued on September 30, 2003. Fimoff is prior art under at least pre-AIA §102(b).

Fimoff first discusses the MPEG-2 standard—a precursor to MPEG-4 and HEVC—that was dominant in Fimoff's time. Ex-1003, ¶¶101-02. At that time, television networks were transitioning from standard-definition signals to high-definition signals (*id.*), which was enabled in-part by MPEG-2's ability to encode video "over a wide range of resolutions, including higher resolutions." Fimoff, 1:25-28. Fimoff provides a high-level overview of the MPEG-2 encoding process. *Id.*, 1:48-3:36.

Fimoff recognized that "[i]n order to permit people to use their existing [standard-definition] televisions so as to view HDTV transmitted programs, it [was] desirable to provide a decoder that decodes high resolution MPEG-2 encoded data as reduced resolution video data." *Id.*, 3:37-47. That way, "a down converting decoder would allow the viewing of HDTV signals without requiring viewers to buy expensive HDTV displays." *Id.*

Accordingly, Fimoff discloses "a down converting decoder for down converting and decoding high resolution encoded video for display by a lower resolution receiver." *Id.*, 1:16-19. To accomplish this, Fimoff receives an incoming high-definition video-image signal, which is downsampled and reconstructed as a lower-resolution, standard-definition image. *See, e.g.*, *id.*, 7:66-8:3, Fig. 5; Ex-1003, ¶103. This standard-definition image is then stored in memory as a "reference" that can be used to predict subsequent images in the video. Fimoff, Fig. 5, 18:15-19; Ex-1003, ¶103.

When predicting a subsequent image, however, there is a mismatch between the high-definition images being sent by the encoder and the standard-definition reference images stored in the decoder's memory. *See, e.g.*, Fimoff, 4:5-15, 19:21-27; Ex-1003, ¶104. To account for this mismatch, Fimoff upsamples the standard-definition reference images back into high-definition images so that the high-definition motion vectors provided by the encoder can be associated with the correct pixel(s) of the standard-definition reference image. Fimoff, 18:6-19, 19:22-31, Fig. 7; Ex-1003, ¶105. After applying the motion vectors, the predicted image is then downsampled back to standard definition and combined with an also-downsampled residual to produce the final reconstructed image in standard definition. Fimoff, 20:25-36, 20:60-21:14, 30:34-38; Ex-1003, ¶106. This final, standard-definition, reconstructed image is then saved to memory as the next reference frame, and displayed on legacy televisions. *See, e.g.*, Fimoff, Fig. 5.

As part of the prediction process, Fimoff explains that the decoder performs "prediction up sampling, linear interpolation [for motion compensation], and prediction down sampling…[in] a single operation." *Id.*, 21:15-17. Moreover, the filtering process is performed in both the horizontal (rows of pixels) and vertical (columns of pixels) direction. *Id.*, 30:20-38, Fig. 5; Ex-1003, ¶107.

4. Claim 1

a. Element 1[pre]

To the extent limiting, the combination of Wiegand and Davies renders element 1[pre] obvious.

As to a "decoding method," Wiegand discloses a "[d]ecoding process" that "[o]utputs…decoded samples of the current picture." Wiegand, 84 (§8); Ex-1003, ¶¶223-25.

Wiegand further discloses that its decoding process is "of a binary stream." Specifically, Wiegand explains that the "[i]nputs to this process are NAL units" (Wiegand, 85 (§8.1)), which constitute a "bitstream." Wiegand, 16 (§3.11) (defining "bitstream" as "[a] sequence of bits that forms the representation of coded pictures …. *Bitstream is a collective term used to refer either to a NAL unit stream* or a byte stream.") (emphasis added); *see also id.*, 27 (§6.1); Ex-1003, ¶¶227-28. A

POSITA would have understood that a bitstream is the claimed "binary stream" because "bits" are binary values transmitted together as part of a "stream" of data. Ex-1003, ¶¶226, 229.

Further, Wiegand discloses that its decoding process "reconstruct[s] a current block of a current image from a reference block of a reference image," as claimed. Ex-1003, ¶230. Specifically, Wiegand's decoding process reconstructs a current image on a block-level basis from a reference image by using an "inter prediction mode." Wiegand, 109 (§8.4); Ex-1003, ¶234. Wiegand explains that "inter coding" refers to "[c]oding of a *block*, macroblock, slice, or picture that uses inter prediction." Wiegand, 18 (§3.47); *see also, e.g.*, *id.*, 117 (showing an example inter prediction process using blocks). "Inter prediction," in turn, is defined as "[a] prediction derived from decoded samples of reference pictures other than the current decoded picture." *Id.*, 18 (§3.48); *see also id.*, 19 (§3.88) (defining "reference picture" as "contain[ing] samples that may be used for inter prediction in the decoding process of subsequent pictures[.]"); Ex-1003, ¶234. A POSITA therefore would have understood that Wiegand's "inter prediction mode" refers to the reconstruction of a current block using a block from a reference image because the current block is being "predicted" based on a block from a reference image. Ex-1003, ¶¶231-36.

Wiegand also discloses that reconstructed reference images are stored in a decoded picture buffer. *See, e.g.*, Wiegand, 17 (§3.30) (defining "decoded picture buffer" as "[a] buffer holding decoded pictures for reference…."), 221 (providing an example flow chart of a decoder picture buffer feeding reference frames to the "[d]ecoding process"); Ex-1003, ¶237. Similarly, Davies recognized that reference pictures are stored (in the buffer), and did not propose altering this aspect of Wiegand. Davies, 3 (§3). A POSITA would have understood that Wiegand's decoding process uses a decoded picture buffer to store reconstructed pictures as reference frames. Ex-1003, ¶237.

Although Wiegand discloses a decoding process in which a current block of a current image is reconstructed from a reference block of a reference image, Wiegand does not expressly disclose reconstructing a current block based on "a reference image…at a different size from the size of said current image," as claimed. Ex-1003, ¶238.

Davies, however, proposes amending Wiegand to expressly include this feature. Davies, 4 (§4) (noting "[v]arious modifications to the current Working Draft (WD) [3] are needed," with "[3]" being Wiegand); Ex-1003, ¶239. Specifically, Davies recognizes that because "streaming [video] over packet[] networks frequently require that the encoded stream adapt to changing network conditions," Davies, 2, Wiegand should be amended to allow different-sized frames to be sent from the encoder to the decoder to allow the system to "adapt seamlessly to adverse network conditions." *id.*, 1, 2 (§1). In other words, Davies teaches a system that would transmit full-resolution frames when there is enough network capacity available and lower-resolution frames when network capacity is constrained (because lower-resolution frames require less data). Ex-1003, ¶¶240-41.

But because a decoder in "inter prediction mode" uses already-reconstructed reference frames to reconstruct new frames, under Davies's proposal, a smaller frame may be used to reconstruct a bigger frame, and vice versa. *Id.*, ¶¶242-45. Davies therefore proposes further modifying Wiegand to allow frames to be "predicted across resolutions by re-scaling reference pictures." Davies, Abstract; Ex-1003, ¶¶243-44. More particularly, Davies proposes using "separable down- and up-scaling filters…for prediction" (Davies, 4 (§3.2)) so that smaller reference frames would be "upscaled" to increase their size, and bigger reference frames would be "downscaled" (e.g., downsampled) to decrease their size, to match the size of the current frame being decoded. Ex-1003, ¶¶244-45.

A POSITA would have been motivated to modify Wiegand's inter-prediction decoding process to receive different sized frames, and then to resize (i.e., resample) those being used as reference frames as needed to reconstruct subsequent, different-sized frames, as taught by Davies. Ex-1003, ¶246. This was the express purpose of Davies. *See* Davies, 4 (§4), 11 (expressly proposing to modify Wiegand as described). Video quality could therefore be maximized without undue delay, even in "adverse or varying network conditions." Davies, 2 (§1); Ex-1003, ¶246.

A POSITA would have had a reasonable expectation of success in making this modification because Davies had already proposed the exact changes needed to Wiegand's system. Davies, 7. These changes would have involved computer coding to operate on simple data structures, such as arrays, which would have well been within the skill of a POSITA. Ex-1003, ¶247.

b. Element 1[a]

The combination of Wiegand, Davies, and Fimoff renders element 1[a] obvious.

To start, although the claims use abbreviations "GFH" and GFv," the specification explains that these are just shorthand for a "single filter," one applied in the horizontal (GFH) direction, and another applied in the vertical (GFv) direction. *See, e.g.*, Ex-1001, 5:58-62 ("[T]he horizontal filters MCIF$_H$ and SCF$_H$ are grouped into a single filter GF$_H$," and same for vertical); Ex-1003, ¶¶248-49.

As discussed in Section V.B.4.a (1[pre]), combining Wiegand and Davies would have allowed video streams to adapt to changing network resource availability. Ex-1003, ¶250. But those in the art were also aware of another problem: that the buffer used to store reference frames is limited and at risk of being overloaded. *Id.*, ¶¶251-254. Although Davies describes sending a decoder different-sized frames and then resizing them as needed when used as reference frames, Davies did not describe in detail how the reference frames are stored at the decoder. *Id*.

Upon combining Wiegand and Davies, a POSITA would have recognized that Davies's resampling process would not alter Wiegand's separately disclosed motion-compensation-interpolation process; the two operations would be performed separately. *See* Davies, 7 (§4.4.1.2) (modifying only §8.4.2.2.1 of Wiegand); Wiegand, 124-30 (§8.4.2.2.2) (describing motion-compensation-interpolation process); Ex-1003, ¶251. Thus, although the combination would have already had all the pieces necessary to predict current frames based on different-sized reference frames (i.e., a resampling filter and motion-compensation-interpolation filter), the combination would not solve the known problem of limited buffer capacity. Ex-1003, ¶254. A POSITA would therefore have looked for additional ways to modify Wiegand and Davies to avoid Davies's resampling operation from further overloading the buffer. *Id.*, ¶255.

A POSITA would have recognized, however, that controlling buffer size when decoding a current frame based on a different-sized reference had already been solved in older video codecs. *Id.* Fimoff, for instance, adapted MPEG-2 to display high-definition video content on legacy, standard-definition displays. Fimoff, 3:37-47; Ex-1003, ¶255. Fimoff was thus also concerned with resizing video images based on resource availability. Fimoff, Abstract ("An MPEG encoded interlaced high definition video stream is decoded and down converted to lower resolution by an MPEG decoder *with a reduced amount of reference picture memory.*") (emphasis added); Ex-1003, ¶¶261, 263.

Fimoff recognized that, because the sizes of reference and current frames differed, the reference frame would need to be resampled to match the size of the current frame for prediction purposes. *See, e.g.*, Fimoff, 4:16-27; Ex-1003, ¶256. However, to avoid storing resampled reference frames in memory, Fimoff combined the resampling *and* motion-compensation-interpolation filter operations into one. Fimoff, 21:15-17 ("prediction up sampling, linear interpolation [(for motion compensation)], and prediction down sampling can be combined into a single operation."), 24:24-28; *see also id.*, 2:1-13 (explaining "linear interpolation" refers to the motion compensation process); 5:21-26 (noting others had combined these operations as well); Ex-1003, ¶¶257-61.

To accomplish this, Fimoff describes how the decoder receives full-resolution (high-definition) motion vectors and a residual signal from the encoder, consistent with conventional decoders. *See, e.g.*, Fimoff, 2:7-19, Fig. 5; Ex-1003, ¶257. The decoder then converts the full-resolution motion vectors to low-resolution (Fimoff, 30:11-24) and locates pixels in the low-resolution reference image that are needed for prediction. Ex-1003, ¶258. Once identified, the relevant pixel regions of the low-resolution reference image are pulled from memory and upsampled to high definition so that the full-resolution motion vectors received from the encoder can be applied to the correct pixels in the (now upsampled) reference image. *See* Fimoff, 18:6-19, 19:22-31, Fig. 7; Ex-1003, ¶¶258-59. But because Fimoff wants to always output a standard-definition video, the predicted block is downsampled again to produce a standard-definition output. *See, e.g.*, Fimoff, 21:10-14, Fig. 7; Ex-1003, ¶260. Fimoff explains that this process uses two different filters—a horizontal filter and a vertical filter—that apply to pixels in a given direction. *See, e.g.*, Fimoff, 30:31-34; Ex-1003, ¶259.

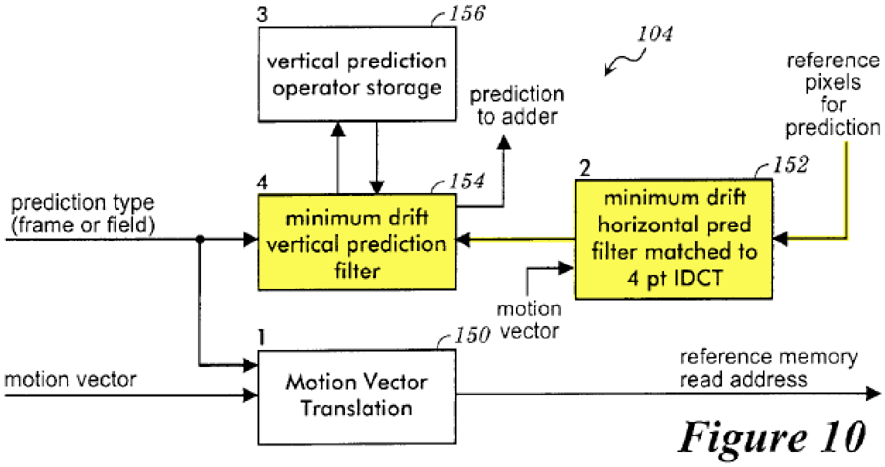


Figure 10

In other words, Fimoff's motion compensator (104) performs both resizing and motion-compensation-interpolation using a single filter in at least one (vertical) direction, while storing only one size of the reference image in memory. Ex-1003, ¶260. And Fimoff explains how the single, combined vertical filter would be applied successively on the columns of pixels of the reference block, as claimed. *See, e.g.*, Fimoff, 30:31-34 (explaining the "vertical filter operator [PL] is…applied to each column of pixels as they are received from the horizontal prediction filter"). Although Fimoff does not expressly say so, a POSITA would have known that Fimoff's horizontal filter is applied successively to lines of pixels as well because this was conventional for horizontal filters. Ex-1003, ¶259; *see, e.g.*, Sita, ¶123; Ex-1050, ¶30 ("applying a horizontal deblocking filter to each set 630 of eight horizontally arranged pixels"), Fig. 6C.

Accordingly, a POSITA would have been motivated to modify Wiegand and Davies's decoder to combine its separate resampling and motion-compensation-interpolation filter operations into a "single filter," like Fimoff, to avoid overloading the buffer with multiple versions of the same reference frame at different sizes. Ex-1003, ¶¶262-63, 265. By combining the two, computational resources would be conserved by avoiding the routing of a decoded signal through separate resampling and motion compensation interpolation filters and/or to/from memory multiple times. *Id.*, ¶263.

A POSITA would have also had a reasonable expectation of success in making this modification because Fimoff provided a description and equations to implement the resampling and motion-compensation filters as one—which simply involve combining separate, known mathematical processes into a single mathematical operation. Ex-1003, ¶264. In other words, the modification would have required only familiarity with computer programming to implement a single, combined mathematical operation to calculate a resampled *and* motion-compensation interpolated reference image, which would have been within the ordinary skill level of those in the art. *Id.*

c. Element 1[b]

Wiegand discloses, or at least renders obvious, element 1[b].

Wiegand explains that a "residue" or "residual" is just the "difference between a prediction of a sample or data element and its decoded value." Wiegand, 20 (§3.95); Ex-1003, ¶¶266-68. In other words, the residue captures any remaining differences between a predicted image and the actual image, which makes the reconstructed image more accurate. Ex-1003, ¶267. To that end, Wiegand discloses a "[d]ecoding process for coding units coded in inter prediction mode consists of…[t]he decoding process for the residual signal of coding units coded in inter prediction mode[.]" Wiegand, 109 (§8.4); Ex-1003, ¶268. Wiegand further provides details and examples regarding how the residue is decoded. *See* Weigand, 130-33 (§8.4.3); Ex-1003, ¶268.

Moreover, as discussed above, the use of residue blocks in image decoding and reconstruction was well-known, and the '877 Patent did not improve upon it. *See* §V.A.3.c (Ground 1, 1[b]). Accordingly, a POSITA would have appreciated that Wiegand teaches the use of residuals in image decoding. Ex-1003, ¶¶269-70.

d. Element 1[c]

The combination of Wiegand, Davies, and Fimoff renders element 1[c] obvious.

As explained in Sections V.B.4.a (1[pre]) and V.B.4.b (1[a]), the combination of Wiegand, Davies, and Fimoff teaches generating a predicted, "motion compensated reference block" as claimed by resampling and motion compensating a block from a different-sized reference image. *See also* Ex-1003, ¶¶271-72. And as explained in Section V.B.4.c (1[b]), Wiegand discloses decoding a residual block of the current block. *See also id.*

For element 1[c], Wiegand discloses reconstructing a current block from a residual block and a motion-compensated reference block. In Wiegand, to reconstruct an image, the predicted, motion-compensated reference block is first determined (pursuant to §§8.4.1 and 8.4.2). Wiegand, 109 (§8.4, steps 1); *see also id.*, 110 (§8.4.1 directing to follow §8.4.2); *see also, generally, id.*, 111-30 (describing calculation of motion compensation). Next, the residue of the block is decoded (pursuant to §8.4.3). *Id.*, 109 Finally, a picture-reconstruction process is followed. *Id.* (step 4); *see also* Ex-1003, ¶273. This is consistent with Fimoff's reconstruction process. *See* Fimoff, 30:34-38 (explaining how a current block is reconstructed by combining the motion-compensated reference block that was output by the horizontal/vertical filters with a residual block); Ex-1003, ¶274.

A POSITA would have understood that a current block is reconstructed by adding the residue block to the predicted, motion-compensated reference block to account for any "difference between a prediction…and its decoded value." Wiegand, 20 (§3.95); Ex-1003, ¶¶273-74. This reconstruction process—combining a motion-compensated reference block and a residue block—was well known before the '877 Patent, and a POSITA would have found it obvious to continue using that process when basing predictions on reference images of different sizes than the current block. Ex-1003, ¶274.

e. Element 1[d]

The combination of Wiegand, Davies, and Fimoff renders element 1[d] obvious.

As explained in Section V.B.4.b (1[a]), the combination of Wiegand, Davies, and Fimoff teaches "applying…a single vertical filter GFv successively…on the columns of pixels of said reference block" as claimed. Section V.A.3.e (Ground 1, 1[d]) further provides a discussion of the meaning of this claim element. Fimoff teaches a single vertical filter (154) comprising a vertical motion compensation interpolation filter (MCIFv) and a vertical resampling filter (SCFv)), with the resampling filter applied first as part of a joint operation. Fimoff, 20:60-64 ("up sampling must be followed by ½ pixel linear interpolation"), 21:52-54 (same), 21:15-17 ("Vertical prediction up sampling, linear interpolation, and prediction down sampling can be combined into a single operation”), 21:55-60 (same); *see also id.*, 20:1-24, 22:47-65, 23:5-24:5, 24:6-55, 30:25-38; *see also generally id.*, 20:31-21, 21:28-67; ’877 Patent, 5:57-67; Ex-1003, ¶¶275-77.

Even though Fimoff teaches the use of a last-stage downsampling filter to convert the predicted frames from a high-definition to a standard-definition format to be displayed on standard-definition televisions (Fimoff, 21:10-27), a POSITA would have further understood that the additional downsampling operation (term d0 in Equation 66) would be unnecessary in the combination of Wiegand and Davies. Ex-1003, ¶278. Wiegand and Davies concern adaptively resizing frames throughout a video sequence in response to changing network conditions, not outputting a consistent, standard-definition video like Fimoff. *Id.* Therefore, a POSITA would have known that Wiegand and Davies’s combined decoder would perform a resampling operation only to align the motion vectors of a current frame with the pixels of a reference frame if the two were different sizes; Fimoff’s final downsampling operation would be unnecessary. *Id.*

A POSITA would have understood that these filters are “applied jointly” in the context of the ’877 Patent (which appears only in the claims). *See* Ex-1001. During prosecution, the Applicant used the phrase “applied jointly” to mean “applied using a single filter.” Ex-1002, 281; Ex-1003, ¶277. Fimoff teaches that a single vertical filter would apply a resizing filter operation and motion compensation interpolation filter operation jointly as part of a "single operation." *See* Fimoff, 21:15-17, 21:55-60.

And for the reasons discussed above, it would have been obvious to combine Wiegand's and Davies's separate filters into one, just like Fimoff. *See* §V.B.4.b (1[a]); Ex-1003, ¶279.

f. Element 1[e]

The combination of Wiegand, Davies, and Fimoff renders element 1[e] obvious.

Element 1[e] recites a substantially similar limitation as element 1[d], but describes the particulars of the horizontal filter (first referenced in 1[a]) instead of the vertical filter (1[d]). As explained in Section V.B.4.e (1[d]), Fimoff teaches the recited requirements for the single vertical filter. A POSITA would have also understood, or found it obvious, that those same teachings apply to the horizontal filter. Ex-1003, ¶¶280-94.

Specifically, Fimoff describes that "[h]orizontal prediction filtering occurs in the horizontal prediction filter 152." Fimoff, 30:22-25. Fimoff also shows a "horizontal prediction filter" (152) in Figure 10, which is reproduced below:

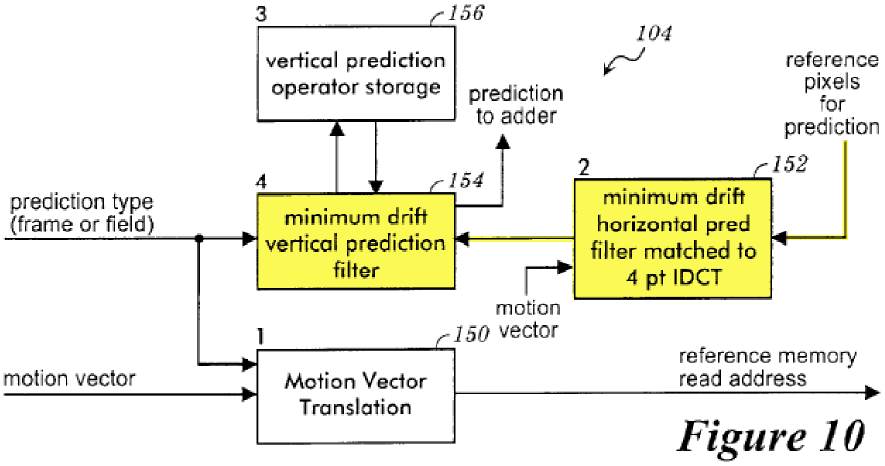


Figure 10

Although Fimoff describes its *vertical* filter in substantial detail—including its inclusion of a resizing and motion compensation interpolation operation—Fimoff provides less detail about the inner workings of the horizontal filter. Ex-1003, ¶281. But a POSITA would have known—or at least found it obvious—that Fimoff's horizontal filter would perform a resampling and interpolation operation as well to avoid distorting an image by resampling it in only one direction and accounting for motion-vector components in only one (vertical) direction. Ex-1003, ¶¶283-89.

As such, a POSITA would have understood that not only could Wiegand's and Davies's separate resampling and motion-compensation-interpolation filter operations be combined, as taught by Fimoff, but that they could also be applied in a vertical and horizontal direction separately, as also suggested by Fimoff. Ex-1003, ¶¶285-89; *see* §V.B.4.e (1[d]). Performing these operations as separate horizontal and vertical filters—also known as one-dimensional (1D) filters—was known in the art. Ex-1003, ¶¶290-94. A POSITA would have also been motivated to apply a joint filter in the horizontal direction as well to avoid having to store resized images in the buffer between the resizing and motion-compensation operations, as discussed in Section V.B.4.b (1[a]).

And for the reasons discussed above, it would have been obvious to combine Wiegand's and Davies's separate filters into one, just like Fimoff. *See* §V.B.4.b (1[a]); Ex-1003, ¶294.

g. Element 1[f]

The combination of Wiegand, Davies, and Fimoff renders element 1[f] obvious.

As explained in Section V.B.4.a (1[pre]), Wiegand discloses a "decoded picture buffer" to store reference frames. *See also* Ex-1003, ¶¶295-96. However, as discussed in Section V.B.4.b (1[a]), limited buffer memory was a known problem, and a POSITA would have been motivated to apply Fimoff's teachings to the combination of Wiegand and Davies to address that problem. *Id.*, ¶296. Moreover, a POSITA would have recognized that combining the resampling/motion-compensation filter operations would achieve "reduced memory in [the] decoder loop." Fimoff, Title; Ex-1003, ¶¶261-62, 296.

Specifically, Fimoff discloses a decoder that includes "memory 106." Fimoff, 30:15-22. A POSITA would have understood this "memory" constitutes a "decoded picture buffer" because it serves the same function. Ex-1003, ¶297. Furthermore, during prosecution of the '877 Patent, the Applicant had analogized the claimed "decoded picture buffer" to a prior-art "reference frame store" (Ex-1002, 237), which the prior art defined as "any type of memory or data storage device to store video blocks reconstructed from previously encoded blocks" (Ex-1017, 10:3-6; Ex-1003, ¶297).

Although the data entering the decoder represents high-definition images, the decoder is not reconstructing frames in high definition and therefore does not store them that way. *See* Fimoff, Fig. 5 (showing only "[r]educed resolution reference pictures" stored in memory 106). Instead, during reconstruction, the images undergo a downsampling operation to reduce them to standard-definition before they are stored in memory as reference frames. *Id*. Figure 5 of Fimoff (annotated) is reproduced below:

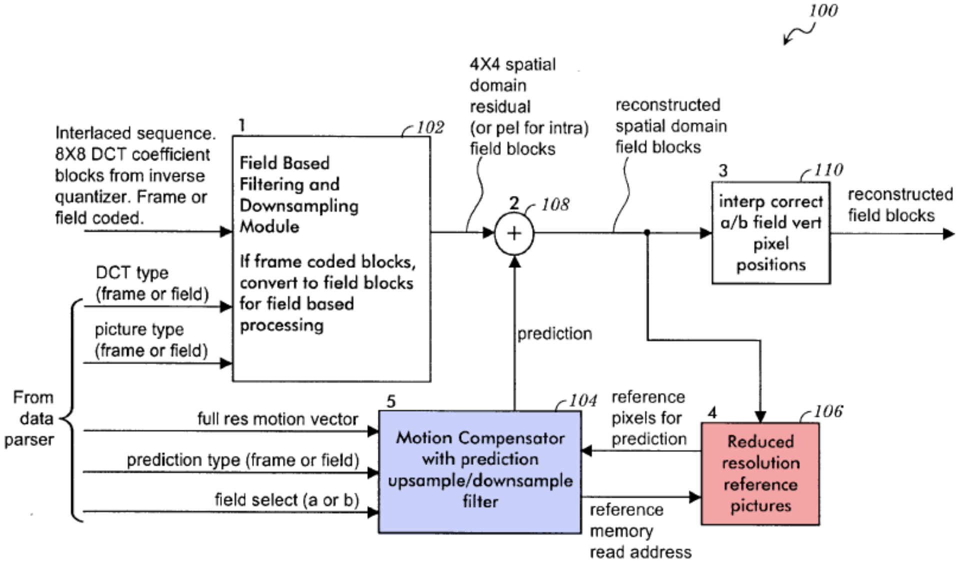


Figure 5

As Figure 5 shows, the combined resampling/motion-compensation-interpolation filter (104) obtains the "reduced resolution reference pictures" from memory (106), which is a "reconstructed reference image" as claimed (because it is an image that has been reconstructed and stored in memory as a reference). Ex-1003, ¶298. When predicting the current (high-definition) frame from the "reduced resolution reference picture," the relevant pixels from the reference picture are upsampled in the combined resampling/motion-compensation-interpolation filter (within 104, *see* Fimoff Fig. 10), without ever storing the *upsampled* version of the "reduced resolution reference picture" in memory. Ex-1003, ¶298. Instead, after upsampling, applying the motion vectors, and performing a final downsampling operation, the output of the combined resampling/motion-compensation-interpolation filter (104) is a predicted standard-definition image, which is added to a standard-definition residual that is output by a separate downsampling module (102). *Id*. Fimoff explains that only *that* fully reconstructed, downsampled image (after the residual is added) is stored in memory as the next "reduced resolution reference picture." Fimoff, 18:15-17. In other words, only the final, "reconstructed reference images are stored in…memory," just as the '877 Patent describes. Ex-1001, 5:44-52; Ex-1003, ¶¶297-98. The combined resampling/motion-compensation filter (104) never provides a different-sized version of the existing reference frame to memory. Ex-1003, ¶298.

Accordingly, a POSITA would have recognized that not only did Fimoff not store any resized versions of a reference image in memory, but there was also no need to do so because the resampling and motion-compensation-interpolation operations are combined. Ex-1003, ¶299. A POSITA therefore would have understood that, upon modifying Wiegand and Davies to combine these operations (like Fimoff) "no resampled version of [a] reconstructed reference image is stored in the decoded picture buffer," as claimed. *Id.*

5. Claim 4

The combination of Wiegand, Davies, and Fimoff renders this claim obvious as a whole.

Wiegand's working draft of the HEVC standard (like the final draft) focuses on the decoder, not the encoder. *See* Wiegand, 5-8 (dedicating §8 to the "Decoding process," but no similar section for the encoding process); Ex-1003, ¶¶300-01. That is because coding standards provide the syntax for bitstreams and how those bitstreams need to be decoded—ensuring that any standard-compliant incoming signal could be decoded by any standard-compliant decoder. Ex-1003, ¶301. Exactly how the standard is implemented at the encoder is less relevant, as long as the encoder produces as its output a signal that fits within the specifications of the decoder in the standard. *Id.*

Encoders typically perform many of the same processes as the decoder, just in reverse. Ex-1003, ¶302; *see also, e.g.*, Fimoff, 2:47-50 ("The decoding process implemented by the MPEG-2 decoder 30 can be thought of as the reverse of the encoding process implemented by the MPEG-2 encoder 10."). In other words, where the encoder takes an image and applies a series of operations to convert the image into a bitstream, the decoder takes the bitstream and reverses the encoder's operations to reconstruct the image again. Ex-1003, ¶¶302-03. In fact, the prior art had long recognized that having the encoder and decoder produce identical reference images was critical for predictive coding; otherwise, decoders risked "drifting" further from the prediction parameters the encoder was using, producing poor reconstruction results. Ex-1003, ¶304; *see, e.g.*, Ex-1032, 1 ("the input signals to the predictor at the encoder and the reduced resolution decoder can diverge [called 'drift'], resulting in a loss of signal quality."); Fimoff, 7:62-65 (building in "minimum drift prediction filtering" operations to avoid drift).

Accordingly, because Wiegand, Davies, and Fimoff collectively provide a process for decoding a bitstream to reconstruct a block of a current image based on a different-sized reference block, implementing this process in reverse at the encoder would been obvious to a POSITA. Ex-1003, ¶¶305-07.

a. Element 4[pre]

To the extent limiting, the combination of Wiegand and Davies render element 4[pre] obvious. *See* §§V.B.4.a (1[pre]), V.B.5 (claim 4, generally); Ex-1003, ¶¶308-09.

b. Element 4[a]

Generally, the combination of Wiegand, Davies, and Fimoff render element 4[a] obvious. *See* §V.B.4.b (1[a]). But the implementation of Wiegand, Davies, and Fimoff at the encoder requires one further consideration.

Specifically, Wiegand and Davies together describe a decoder that can process different-sized frames sent from an encoder in response to changing network conditions. *See* §V.B.4.a (1[pre]); Ex-1003, ¶311. And Fimoff teaches using a combined resampling/motion-compensation-interpolation operation when a decoder receives different-sized reference images at the decoder, which a POSITA would have recognized would reduce the burdens on Wiegand's image buffer. *See* §V.B.4.b (1[a]); Ex-1003, ¶¶310-12.

Although Fimoff focused on the decoder so that incoming high-definition signals would be consistently converted to standard-definition images to support legacy television sets, a POSITA would have appreciated that, when combined with Wiegand, it would have been further obvious to combine the resampling/motion-compensation-interpolation operations at the encoder as well. Ex-1003, ¶313.

A POSITA would have understood that encoders are essential for predictive coding to calculate the motion vectors and residual in the first instance. *See generally*, Ex-1003, ¶¶44-45, 49, 314; *see, e.g.*, Fimoff, 1:48-64 (describing how an MPEG-2 encoder contains a "motion estimator [that] compares each new macroblock to be encoded with the macroblocks in a reference picture previously stored in a reference picture memory 16," and subtracts the predicted block from the original image to calculate the residual). But unlike Fimoff, an encoder consistent with Wiegand and Davies would not consistently transmit the same (high-definition) sized images, and would instead adaptively resize images *at the encoder* to adapt to changing network conditions. Ex-1003, ¶313. For this reason, just like the decoder, the encoder would need to encode current images using different-sized reference images when the encoder needs to change resolutions in response to network conditions. Ex-1003, ¶¶313-14.

A POSITA also would have appreciated that the buffer memory at the encoder is limited, just like the decoder. Ex-1003, ¶313. A POSITA therefore would have further been motivated to incorporate Fimoff's combined resampling/interpolation filter for the same reasons a POSITA would use Fimoff's combined filter at the decoder: to avoid overloading that buffer. Ex-1003, ¶314; *see* §V.B.4.b (1[a]). Moreover, it was understood to use as close to the same filtering operations as possible at both the encoder and decoder to avoid drift. Ex-1003, ¶314.

For this reason, it would have been obvious to not only combine the resampling/motion-compensation-interpolation filter operations at the decoder (like Fimoff), but also at the encoder when implementing the combination of Wiegand and Davies. Ex-1003, ¶¶313-14.

c. Element 4[b]

Wiegand discloses, or at least renders obvious, element 4[b].

Wiegand discloses that, for encoding, a "prediction residual" is calculated after estimating and accounting for the motion vectors. Wiegand, 14. A POSITA would have understood that Wiegand's residue is therefore calculated from the current block and the predicted, motion-compensated reference block because that is how residuals are conventionally determined in predictive coding. Ex-1003, ¶¶315-16; *see also* Fimoff, 1:48-2:6 (explaining how "motion estimator 14 reads this matching macroblock…out of the reference picture memory 16 and sends it to the subtractor 12 which subtracts it…from the new macroblock entering the MPEG-2 encoder 10. The output of the subtractor 12 is an error, or residual, that represents the difference between the predicted macroblock and the new macroblock being encoded.”).

d. Element 4[c]

Wiegand discloses, or at least renders obvious, element 4[c].

Wiegand discloses that, for encoding, a “prediction residual” is “compressed using a transform…before it is quantised,” which is then combined with the motion vectors “and encoded using either variable length coding or arithmetic coding.” Wiegand, 14. A POSITA would have understood that by referring to “variable length coding or arithmetic coding,” Wiegand discloses encoding the residue block into a binary stream because that is how digital data is conventionally processed and encoded. Ex-1003, ¶¶317-20; *see also* Fimoff, 2:13-19 (“The [residual] data encoded by the coder 22 are combined with the motion vector data…and the combined data are transmitted to a receiver that includes an MPEG-2 decoder 30.”).

e. Element 4[d]

The combination of Wiegand, Davies, and Fimoff render element 4[d] obvious. *See* §§V.B.4.e (1[d]), V.B.5 (claim 4, generally); Ex-1003, ¶321.

f. Element 4[e]

The combination of Wiegand, Davies, and Fimoff render element 4[e] obvious. *See* §§V.B.4.f (1[e]), V.B.5 (claim 4, generally); Ex-1003, ¶322.

g. Element 4[f]

The combination of Wiegand, Davies, and Fimoff render element 4[f] obvious. *See* §§V.B.4.g (1[f]), V.B.5 (claim 4, generally); Ex-1003, ¶323.

6. Claim 7

a. Element 7[pre]

To the extent limiting, Wiegand discloses, or at least renders obvious, element 7[pre].

Wiegand discloses that the working draft HEVC standard was intended to address a need "for higher compression of moving pictures for various applications such as videoconferencing, digital storage media, television broadcasting, internet streaming, and communication." Wiegand, 13. Wiegand envisions that it would allow for "motion video to be manipulated as a form of computer data and to be stored on various storage media, [and] transmitted and received over existing and future networks." *Id.* Accordingly, a POSITA would have understood or found it obvious that Wiegand's decoding process was intended for computers, which would have required at least a computer processor and memory that constitutes a circuit. Ex-1003, ¶¶324-27; *see* §IV (Claim Construction).

b. Element 7[a]

The combination of Wiegand and Davies renders element 7[a] obvious. *See* §V.B.4.a (1[pre]); Ex-1003, ¶328.

c. Element 7[b]

The combination of Wiegand, Davies, and Fimoff renders element 7[b] obvious. *See* §V.B.4.b (1[a]); Ex-1003, ¶329.

d. Element 7[c]

Wiegand discloses, or at least renders obvious, element 7[c]. *See* §V.B.4.c (1[b]); Ex-1003, ¶330.

e. Element 7[d]

Wiegand discloses, or at least renders obvious, element 7[d]. *See* §V.B.4.d (1[c]); Ex-1003, ¶331.

f. Element 7[e]

The combination of Wiegand, Davies, and Fimoff renders element 7[e] obvious. *See* §V.B.4.e (1[d]); Ex-1003, ¶332.

g. Element 7[f]

The combination of Wiegand, Davies, and Fimoff renders element 7[f] obvious. *See* §V.B.4.f (1[e]); Ex-1003, ¶333.

h. Element 7[g]

The combination of Wiegand, Davies, and Fimoff renders element 7[g] obvious. *See* §V.B.4.g (1[f]); Ex-1003, ¶334.

7. Claim 8

a. Element 8[pre]

To the extent limiting, Wiegand discloses, or at least renders obvious, element 8[pre]. *See* §§V.B.5 (claim 4, generally) (encoder), V.B.6.a (7[pre]) (circuit); Ex-1003, ¶335.

b. Element 8[a]

The combination of Wiegand and Davies renders element 8[a] obvious. *See* §V.B.4.a (1[pre]), V.B.5 (claim 4, generally); Ex-1003, ¶336.

c. Element 8[b]

The combination of Wiegand, Davies, and Fimoff renders element 8[b] obvious. *See* §V.B.4.b (1[a]), V.B.5 (claim 4, generally); Ex-1003, ¶337.

d. Element 8[c]

Wiegand discloses, or at least renders obvious, element 8[c]. *See* §V.B.5.c (4[b]); Ex-1003, ¶338.

e. Element 8[d]

Wiegand discloses, or at least renders obvious, element 8[c]. *See* §V.B.5.d (4[c]); Ex-1003, ¶339.

f. Element 8[e]

The combination of Wiegand, Davies, and Fimoff renders element 8[e] obvious. *See* §V.B.4.e (1[d]), V.B.5 (claim 4, generally); Ex-1003, ¶340.

g. Element 8[f]

The combination of Wiegand, Davies, and Fimoff renders element 8[f] obvious. *See* §V.B.4.f (1[e]), V.B.5 (claim 4, generally); Ex-1003, ¶341.

h. Element 8[g]

The combination of Wiegand, Davies, and Fimoff renders element 8[g] obvious. *See* §V.B.4.g (1[f]), V.B.5 (claim 4, generally); Ex-1003, ¶342.

C. Ground 3: Obviousness over Wiegand in view of Davies and Sita

Ground 3 differs from Ground 2 in a single respect: it relies on Sita's teachings of the resampling/motion-compensation-interpolation filter rather than Fimoff, because a POSITA would have found either reference in combination with Wiegand and Davies renders the claims obvious. Ex-1003, ¶¶343-44. Accordingly, this Ground addresses only the claim elements related to motion compensation (elements 1[a], 4[a], 7[b], and 8[b]), the particular filters (elements 1[d]-[e], 4[d]-[e], 7[e]-[f], 8[e]-[f]), and the frame memory (elements 1[f], 4[f], 7[g], and 8[g]) because only those elements rely on Fimoff's teachings. To the extent Fimoff was cited in other elements of Ground 2 to support a POSITA's general understanding of the art—*see, e.g.*, §V.B.4.d (citing Fimoff to show the conventional nature of reconstructing current images using a residual)—Sita provides that same support, as discussed with respect to each element in Ground 1.

1. 1[a], 4[a], 7[b], and 8[b]

As discussed in Ground 2, a POSITA would have recognized that the combination of Wiegand and Davies would have resulted in a system that may perform resampling and motion-compensation interpolation separately, such that resampled versions of reference images may need to be stored in the reference buffer that may already be at risk of overloading. *See* §V.B.4.b (Ground 2, 1[a]); Ex-1003, ¶¶345-46. However, a POSITA would have known that, not only did Fimoff already solve this problem, but so had Sita in a similar way. Ex-1003, ¶346.

Specifically, Sita (like Fimoff) teaches a decoder that takes incoming high-definition signals to produce a standard-definition decoded video. *See* Sita, ¶¶2-3, 66-67. And Sita (like Fimoff) uses standard-definition reference images to reconstruct high-definition images (*id.*, ¶¶87-89) and acknowledges one benefit of this is a "considerable reduction of memory required for storing reference images," *id.*, ¶83. And Sita (like Fimoff) teaches a process of resampling reference images to match the resolution of the incoming (high-definition) signal to accurately apply the motion vectors used to predict the current image from the reference. *Id.*, ¶¶88-89. However, (unlike Fimoff) Sita is not concerned with the details of resampling interlaced video, and Sita expressly recognizes the motion-compensation process can be performed by using separate horizontal and vertical filters. *Id.*, ¶95. At base, Sita teaches a process to motion compensate reference blocks that have a different size as the current image by applying a single horizontal and vertical filter to the lines and columns of pixels in the reference block, as claimed. *See* §V.A.3.b (Ground 1, 1[a]); Ex-1003, ¶347.

Accordingly, a POSITA would have been motivated to modify Wiegand and Davies's decoder to combine its separate resampling and motion-compensation-interpolation filters into a "single filter," like Sita, to avoid overloading the buffer with multiple versions of the same reference frame at different sizes. Ex-1003, ¶348. By combining the two, computational resources would be conserved by avoiding the routing of a decoded signal through separate resampling and motion compensation interpolation filters and/or to/from memory multiple times. *Id.*

Once again, a POSITA would have had a reasonable expectation of success in making this modification because Sita already provided all of the equations to implement the filters as one, which would have required only familiarity with computer programming within the ordinary skill level of those in the art. *Id.*, ¶349.

2. 1[d]-[e], 4[d]-[e], 7[e]-[f], 8[e]-[f]

As explained in Sections V.A.3.e (Ground 1, 1[d]) and V.A.3.f (Ground 1, 1[e]), Sita teaches the particular arrangement of a single vertical and horizontal filter, such that each performs a resampling operation and motion-compensation-interpolation operation, applied jointly. And as explained in Section V.C.1, a POSITA would have found it obvious to combine the separate resampling and motion-compensation-interpolation operations taught by Wiegand and Davies to avoid needing to store resampled reference images in the buffer, as taught by Sita. Therefore, a POSITA would have found these elements obvious as well based on the combination of Wiegand, Davies, and Sita. Ex-1003, ¶¶350-51.

3. 1[f], 4[f], 7[g], and 8[g]

As explained in Section V.A.3.g (Ground 1, 1[f]), Sita teaches a process of jointly performing a resampling and motion-compensation-interpolation operation without needing to store a resampled version of the reconstructed reference image in a buffer. Accordingly, a POSITA would have recognized that not only did Sita not store any resized versions of a reference image in memory, but there was also no need for Sita to do so because the resampling and motion-compensation operations are combined in a single operation. Ex-1003, ¶¶352-53. A POSITA therefore would have understood that, upon modifying Wiegand and Davies to combine these operations (like Sita) "no resampled version of [a] reconstructed reference image is stored in the decoded picture buffer," as claimed. *Id*.

VI. CONCLUSION

Inter Partes Review of the Challenged Claims is respectfully requested.

VII. MANDATORY NOTICES

A. Real Party in Interest

Petitioner identifies themselves as real parties in interest.

B. Related Matters

The '877 patent has been involved in the following matters:

- *InterDigital, Inc. v. Lenovo (United States) Inc.*, No. 5-23-cv-00493 (E.D.N.C. Sept. 1, 2023); and
- *Elec. Devices Including Smartphones, Computs., Tablet Computs., and Components Thereof.*; Inv. No. 337-TA-1373 (Violation) (ITC Sept. 5, 2023).

C. Notice of Counsel and Service Information

LEAD COUNSEL
Jessica Kaiser (Reg. No. 58,937) kaiser-ptab@perkinscoie.com PERKINS COIE LLP 1900 Sixteenth Street, Suite 1400 Denver, CO 80202-5255 Telephone: (303) 291-2300

BACK-UP COUNSEL
Robin Brewer, Reg. No. 60,951 PERKINS COIE LLP 505 Howard Street Suite 1000 San Francisco, CA 94105 Tel: (415) 344-7000 Brewer-ptab@perkinscoie.com Jon Carter, Reg. No. 75,145 PERKINS COIE LLP 1155 Avenue of the Americas 22nd Floor. New York, NY 10036-2711 Tel: (212) 262-6900 Carter-ptab@perkinscoie.com Matthew A. Lembo, Reg. No. 75,633 PERKINS COIE LLP 1155 Avenue of the Americas, 22nd Floor New York, NY 10036-2711 Telephone: (212) 262-6900 Lembo-ptab@perkinscoie.com

Petitioner consents to electronic service. All services and communications to the attorneys listed above may be sent to:

Amazon-InterDigital-IPR@perkinscoie.com

D. Power of Attorney

A power of attorney is filed herewith according to 37 C.F.R. §42.10(b).

Respectfully submitted,

/ *Jessica Kaiser* /

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

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

Date: January 9, 2026

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

Pursuant to 37 C.F.R. §42.24(a), Petitioner hereby certifies that portions of the above-captioned Petition for *inter partes* review of U.S. Patent No. 10,250,877, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 13,703. Pursuant to 37 C.F.R. §42.24(a), this word count is in compliance and excludes the table of contents, table of authorities, mandatory notices under §42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing. This word count was prepared using Microsoft Word.

Respectfully submitted,

/ *Jessica Kaiser* /

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

Date: January 9, 2026
PERKINS COIE LLP
1900 Sixteenth Street, Suite 1400
Denver, CO 80202-5255

CERTIFICATE OF SERVICE

The undersigned hereby certifies that true copies of the Petition for *inter partes* review of U.S. Patent No. 10,250,877 and supporting materials (Exhibits and Power of Attorney) were served via overnight delivery on the Patent Owner at the correspondence address of record as listed on Patent Center:

Patent Docketing
200 Bellevue Parkway, Suite 300
Wilmington, DE 19809

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

Alan Lee Whitehurst (awhitehurst@mckoolsmith.com)
Richard Alan Kamprath (rkamprath@mckoolsmith.com)

Respectfully submitted,

/ *Jessica Kaiser* /

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

Date: January 9, 2026