

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SNAP INC. and HISENSE USA CORPORATION,
PETITIONERS

v.

NOKIA TECHNOLOGIES OY,
PATENT OWNER.

IPR2025-01014
U.S. Patent No. 8,050,321

**DECLARATION OF DR. JOSEPH P. HAVLICEK
IN SUPPORT OF PETITION FOR
INTER PARTES REVIEW OF U.S. PATENT NO. 8,050,321**

TABLE OF CONTENTS

I.	INTRODUCTION AND COMPENSATION	1
II.	BACKGROUND AND QUALIFICATIONS	2
A.	Educational Background.....	2
B.	Professional Experience.....	3
C.	Publications.....	7
III.	LEVEL OF ORDINARY SKILL IN THE ART	8
IV.	MATERIALS REVIEWED	9
V.	LEGAL STANDARDS.....	11
A.	Priority Date.....	11
B.	Claim Interpretation	12
C.	Prior Art	13
D.	Obviousness	13
VI.	SUMMARY OF OPINIONS	16
VII.	OVERVIEW OF THE '321 PATENT.....	17
A.	Technology Overview.....	17
B.	The Challenged Claims.....	21
VIII.	CLAIM CONSTRUCTION	22

IX.	DETAILED EXPLANATION OF GROUNDS	22
A.	Ground 1: Kim Renders Obvious Claims 1, 3-4, 6-8, 10, and 11	22
1.	Kim	22
2.	Claim 1.....	28
3.	Claim 3: “A method according to claim 1, further comprising: encoding into the video sequence an identifier value for the independent sequence.”	38
4.	Claim 4.....	39
5.	Claim 6: “A video encoder according to claim 4, wherein the video encoder is configured to encode into the video sequence an identifier value for the independent sequence.”	41
6.	Claim 7.....	42
7.	Claim 8.....	45
8.	Claim 10.....	63
9.	Claim 11.....	67
B.	Ground 2: Kim in view of Adolph Render Obvious claims 2, 5, and 9 ..	73
1.	Adolph	73
2.	Motivation to Combine Kim and Adolph.....	76

3.	Claim 2: “A method according to claim 1, further comprising: encoding the indication into the video sequence as a separate flag included in the header of a slice.”	79
4.	Claim 5: “A video encoder according to claim 4, wherein the video encoder is configured to encode the indication into the video sequence as a separate flag included in the header of a slice.”	80
5.	Claim 9: “A method according to claim 8, wherein the indication is a separate flag included in the header of a slice.”	80
X.	CONCLUSION	81

I, **Dr. Joseph P. Havlicek**, hereby declare as follows:

I. INTRODUCTION AND COMPENSATION

1. I have been retained by Hisense and Snap (“Petitioners”), as an independent expert consultant in this *inter partes* review (“IPR”) proceeding before the United States Patent and Trademark Office (“PTO”).

2. I have been asked by Counsel for Petitioners to consider whether certain references teach or suggest the features cited in claims 1-11 of U.S. Patent No. 8,050,321 (“the ’321 patent”) (EX1001). My opinions and the bases for those opinions are set forth below. This declaration is based in part on a prior declaration I prepared in early 2024 for Amazon.com, Inc. in IPR Case No. IPR2024-00691 involving the same Nokia ’321 patent. The prior Amazon IPR was instituted but I understand the proceeding was terminated based on a settlement between Nokia and Amazon.

3. I am being compensated for my time at my standard consulting rate. I am also being reimbursed for expenses that I incur during the course of this work. My compensation in no way depends on the outcome of this matter. I have no other financial interest in this proceeding.

4. My analysis here is based on my years of education, research and experience, as well as my investigation and study of relevant materials, including those cited herein. I may rely upon these materials, my knowledge and experience, and/or additional materials to rebut arguments raised by the Patent Owner. Further, I may also

consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

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

II. BACKGROUND AND QUALIFICATIONS

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

7. My qualifications are set forth in my curriculum vitae, a copy of which is filed as Exhibit EX1004. The following is a brief summary of my relevant qualifications and professional experience.

A. Educational Background

8. I received a Bachelor of Science Degree in electrical engineering with minors in mathematics and computer science from Virginia Tech in 1986. I also received a Master of Science degree in electrical engineering from Virginia Tech in 1988. I received a Ph.D. in Electrical and Computer Engineering from the University of Texas at Austin in 1996. My Ph.D. research was in the field of image processing.

B. Professional Experience

9. From December 1984 to May 1987, I was a software engineer at Management Systems Laboratories, Blacksburg, VA. My main job responsibilities included developing software for nuclear materials management under contract with the United States Department of Energy.

10. From June 1987 to January 1997, I was an electrical engineer at the United States Naval Research Laboratory. For the period of June 1987 through August 1989, I was an on-site contractor affiliated with SFA, Inc., Landover, Maryland. From August 1989 through January 1997, I was a regular government employee. I was on leave without pay from August 1987 through July 1988 while completing my Master of Science degree. I was also on leave without pay for much of the period from August 1990 through January 1997 while I completed my Ph.D. My main job responsibilities at the United States Naval Research Laboratory included designing digital and analog circuits to process real-time video signals and designing and implementing target detection, tracking, and identification algorithms. I developed missile warning receivers for Navy aircraft including the Navy's first two-color infrared system. I was a recipient of the 1990 Department of the Navy Award of Merit for Group Achievement for this work. The production version of this system was deployed on Navy helicopters and saw combat in Afghanistan and Iraq.

11. From January 1993 through December 1993, I was an on-site contractor at International Business Machines (IBM) Corporation, Austin, TX. My main job

responsibilities included designing and implementing image compression and decompression algorithms (CODECs) for IBM products.

12. Since January 1997, I have been a regular faculty member in the School of Electrical and Computer Engineering at the University of Oklahoma, Norman, OK. I was an Assistant Professor from January 1997 through June 2002. I was promoted to the rank of Associate Professor and granted tenure in July 2002. I was promoted to the rank of Professor in July 2007. I was appointed to the Williams Companies Foundation Presidential Professorship in April 2009. In April 2017 I was appointed to the Gerald Tuma Presidential Professorship. My main job responsibilities at the University of Oklahoma include conducting academic research in electrical and computer engineering, teaching graduate and undergraduate courses in electrical and computer engineering, and performing professional and institutional service.

13. I am a member of several professional societies and organizations, including the Institute of Electrical and Electronics Engineers (IEEE), the IEEE Signal Processing Society, the IEEE Computer Society, and the IEEE Intelligent Transportation Society. I am a Senior Member of the IEEE. From November 2015 through February 2018, I served as a Senior Area Editor for the IEEE Transactions on Image Processing. I was formerly an Associate Editor for the IEEE Transactions on Image Processing from December 2010 through October 2015. I served as a Technical Area Chair for the IEEE International Conference on Image Processing in the area of

Image & Video Analysis, Synthesis, and Retrieval (2012, 2013) and have served on the organizing committee of that conference (2007). I have also served as a Technical Area Chair for the IEEE International Conference on Acoustics, Speech, and Signal Processing in the area of Image, Video, and Multidimensional Signal Processing (2012-2014).

14. For over 35 years, I have conducted research and taught classes in the field of image processing and image analysis. My main scholarly contributions have been in the areas of modulation domain image models and image processing (AM-FM image models), video target tracking, and distributed control of video networks for intelligent transportation systems. I have supervised or co-supervised 12 Ph.D. students to completion and I am currently supervising two Ph.D. students. I have been a member of 67 additional doctoral dissertation committees. I have supervised 27 M.S. students to completion, and I am currently supervising two additional M.S. students. I have served as a committee member on 68 additional M.S. committees.

15. I teach a variety of courses at the University of Oklahoma, including the required junior-level Signals and Systems course ECE 3793 (taught 21 times), the graduate level Digital Image Processing course ECE 5273 (taught 23 times), and the graduate level Digital Signal Processing course ECE 5213 (taught 17 times). I have received a number of teaching awards, including the University of Oklahoma College of Engineering Outstanding Faculty Advisor Award (2005-2006) and the University of

Texas Engineering Foundation Award for Exemplary Engineering Teaching while Pursuing a Graduate Degree (1992).

16. Since joining the University of Oklahoma in January 1997, I have been Principal Investigator or Co-Principal Investigator on over 110 externally funded grants and contracts with a total value of \$27,061,653. My main research contributions have been in the areas of signal, image, and video processing, video target tracking, and intelligent transportation systems. I have been author or co-author on over 130 scholarly publications in these areas. I was a recipient of the 1990 Department of the Navy Award of Merit for Group Achievement for my work in video target tracking. My research group at the University of Oklahoma originated the Virtual Traffic Management Center concept featured in a December 2014 FHWA technical report (Guidelines for Virtual Transportation Management Center Development) and a November 2014 FHWA national webinar with the same title. I am co-founder and director of the University of Oklahoma Center for Intelligent Transportation Systems. Under my supervision, the Center has collaborated with the Oklahoma Department of Transportation since 1998 to design and implement the Oklahoma Statewide Intelligent Transportation System, including a geographically distributed video network that is currently deployed on major highways and interstates across the entire State of Oklahoma.

C. Publications

17. A complete listing of my publications is found in my curriculum vitae (EX1004). I highlight some of the publications relevant to the subject matter of the '321 patent below:

- J.P. Havlicek, T.N. Arian, H. Soltani, T. Przebinda, and M. Özaydın, “A preliminary case for Hirschman transform video coding,” in Proc. IEEE Southwest Symp. Image Anal. & Interp., Santa Fe, NM, Mar. 29-31, 2020, pp. 104-107.
- E. Vorakitolan, J.P. Havlicek, R.D. Barnes, and A.R. Stevenson, “Simple, Effective Rate Control for Video Distribution in Heterogeneous Intelligent Transportation System Networks,” in Proc. IEEE Southwest Symp. Image Anal. & Interp., San Diego, CA, Apr. 6-8, 2014, pp. 37-40.
- V. DeBrunner, J.P. Havlicek, T. Przebinda, and M. Özaydın, “Entropy-Based Uncertainty Measures for $L_2(\mathbb{R}^n)$, $l_2(\mathbb{Z})$, and $l_2(\mathbb{Z}/N\mathbb{Z})$ with a Hirschman Optimal Transform for $l_2(\mathbb{Z}/N\mathbb{Z})$,” IEEE Trans. Signal Process., vol. 53, no. 8, pp. 2690-2699, Aug. 2005.
- O. Alkhouli, V. DeBrunner, and J. Havlicek, “Hirschman Optimal Transform (HOT) DFT Block LMS Algorithm,” in Adaptive Filtering, L. Garcia, ed., ISBN: 978-953-307-158-9, InTech, Sep. 2011, pp. 135-152.

- V. DeBrunner, M. Özaydın, T. Przebinda, and J. Havlicek, “The optimal solutions to the continuous- and discrete-time versions of the Hirschman uncertainty principle,” in *Proc. IEEE Int’l. Conf. Acoust., Speech, Signal Process.*, Istanbul, Turkey, Jun. 5-9, 2000, vol. 1, pp. 81-84.

III. LEVEL OF ORDINARY SKILL IN THE ART

18. I understand that a person of skill in the art (“POSITA”) is a hypothetical person who is presumed to have known the relevant art at the time of the invention. To determine the level of skill held by a person of ordinary skill in the art, I understand that a number of factors may be considered, including the following:

- The type of problems encountered in the art;
- Prior art solutions to those problems;
- Rapidity with which innovations are made;
- Sophistication of the technology; and
- Educational level of active workers in the field.

19. I understand that not all of the above factors will be present in every case and that one or more factors may predominate.

20. In view of the above factors, it is my opinion that a POSITA at the time of the alleged invention of the ’321 patent would have had a (1) Bachelor’s degree in electrical engineering, computer engineering, computer science, or a comparable field of study, such as physics and (2) approximately two to three years of practical

experience with video encoding/decoding. Additional experience can substitute for the level of education, and vice-versa.

21. As of the January 23, 2002, I met and exceeded the qualifications of a POSITA. My opinions are offered from this perspective, even if they do not specifically refer to the perspective of a POSITA.

IV. MATERIALS REVIEWED

22. My opinions contained in this Declaration are based on the documents I reviewed, my professional judgment, as well as my education, experience, and knowledge.

23. In forming my opinions expressed in this Declaration, I have reviewed and considered the following materials:

Exhibit	Reference
1001	U.S. Patent No. 8,050,321
1002	U.S. Patent No. 8,050,321 File History
1003	Declaration of Dr. Joseph Havlicek
1004	Curriculum Vitae of Dr. Joseph Havlicek
1005	ISO/IEC 11172-2 (1st ed.) Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s - Part 2: Video ("MPEG-1")
1006	U.S. Patent No. 6,912,351 ("Kim")
1007	CL480PC MPEG-1 System Decoder Specification, available at https://web.archive.org/web/19970128081644/http://www.c-cube.com/prdctlst/cl480pc.pdf .
1008	U.S. Patent No. 5,838,265 ("Adolph")

1009	Ruby B. Lee et al., <i>Real-Time Software MPEG Video Decoder on Multimedia-Enhanced PA 7100LC Processors</i> , Hewlett-Packard J., Apr. 1995, available at https://archive.org/details/Hewlett-Packard_Journal_Vol._46_No._2_1995-04_Hewlett-Packard
1010	Ketan Patel, et al., Performance of a Software MPEG Video Decoder, Multimedia '93: Proceedings of the First ACM Int'l Conf. on Multimedia, August 2-6, 1993
1011	Excerpt from Joan L. Mitchell et al., <i>MPEG Video Compression Standard</i> (Chapman & Hall, 1996)
1012	Excerpt from Roy Hoffman, <i>Data Compression in Digital Systems</i> (Chapman & Hall, 1997)
1013	Sony Corporation, Operating Instructions for SVT-S3050P Time Lapse Videocassette Recorder (1997), available at https://pro.sony/s3/cms-static-content/operation-manual/3193023022.pdf
1014	GYJR Inc., GYJR Products Application Guide (1998), available at https://web.archive.org/web/19980628195831/http://www.gyyr.com:80/pdf/appguide.pdf
1015	C-Cube CLM4111 MPEG-1 Video Encoder data sheet, available at https://web.archive.org/web/19970707143632/http://www.c-cube.com/products/CLM4111.pdf
1016	ISO/IEC 13818-2 (1st ed.) Information Technology – Generic coding of moving pictures and associated audio information: Video (“MPEG-2”)
1017	ISO/IEC TR 11172-5 (1st ed.) Information technology – coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 5: Software simulation
1018	Original Complaint, Nokia Techs. Oy v. Hisense Co. Ltd. and Hisense USA Corp., No. 1:25-cv-01871 (N.D. Ga. April 7, 2025), ECF No. 1.
1019	Complaint Under Section 337 of The Tariff Act of 1930, As Amended, Inv. No. 337-TA-3821 (Apr. 11, 2025)
1020	U.S. Patent No. 7,894,521
1021	IPR2024-00691, Paper 10: Decision Granting Institution of Inter Partes Review
1022	Thomas Sikora, MPEG Digital Video-Coding Standards, IEEE SIGNAL PROCESSING MAGAZINE (Sept. 1997)

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

25. My opinions have also been guided by my appreciation of how a POSITA would have understood the claims and the specification of the '321 patent at the time of the alleged invention, which I have been asked to initially consider to be January 23, 2002 (the filing date of Finnish Application No. 20020127). My opinions reflect how one of ordinary skill in the art would have understood the '321 patent, the prior art to the patent, and the state of the art at the time of the alleged invention.

V. LEGAL STANDARDS

26. I am not an attorney and offer no legal opinions, but in the course of my work, I have had experience studying and analyzing patents and patent claims from the perspective of a POSITA.

27. For the purposes of this Declaration, I have been informed about certain aspects of the law that are relevant to my analysis, as summarized below.

A. Priority Date

28. I understand that, to gain benefit of the filing date of an earlier patent application, each application in the chain leading back to the earlier application must comply with the statutory written description requirement.

29. I understand that, to satisfy the written description requirement, a patent's specification must reasonably convey to a POSITA that the inventor had possession of

the claimed subject matter when the patent application was filed. I understand that, to show “possession” of the invention, the disclosure must describe the invention, including all of its claimed limitations. It is not enough that a claimed invention is an obvious variant of that which is disclosed in the specification.

30. I further understand that entitlement to a filing date extends only to subject matter that is disclosed and not to subject matter that is not disclosed but would be obvious over what is expressly disclosed. Thus, a prior application itself must describe an invention, and do so in sufficient detail that a POSITA can clearly conclude that the inventor invented the claimed invention as of the filing date sought.

31. I also understand that a patent’s priority date is a legal determination based on underlying factual issues. For that reason, I am not rendering a legal opinion on the ultimate legal question of priority date. Rather, my testimony addresses the underlying facts and factual analysis that would support a legal determination of the priority date.

B. Claim Interpretation

32. I understand from Counsel that during an IPR proceeding, claims are to be construed in light of the specification as would be read by a POSITA at the time the application was filed. I understand that claim terms are given their ordinary and customary meaning as would be understood by a POSITA in the context of the entire disclosure. A claim term, however, will not receive its ordinary meaning if the patentee acted as his or her own lexicographer and clearly set forth a definition of the claim term

in the specification. In that case, the claim term will receive the definition set forth in the patent.

C. Prior Art

33. I understand from Counsel that certain patents and printed publications can be considered prior art in an IPR proceeding. I understand that prior art must predate the effective filing date of the claims of the patent-at-issue.

D. Obviousness

34. I understand that a claim is unpatentable if the claim limitations, as arranged in a claim, would have been obvious to a person of ordinary skill in the art as of the date of invention. For example, a combination of familiar elements according to known methods is likely to be obvious if the combination does no more than produce predictable results.

35. In determining obviousness, I understand that the analysis may take into account multiple prior art references, as well as the common knowledge possessed by a POSITA (as discussed above). I further understand that the POSITA is not an automaton but one having common sense and ordinary creativity. I also understand that a claim is not obvious simply because a combination of prior art references discloses each claim element collectively.

36. In determining obviousness, I understand that one should consider the following.

- The scope and content of the prior art;

- The differences between the claimed invention and the prior art;
- The level of ordinary skill in the pertinent art; and
- Secondary considerations of obviousness.

37. I understand that a POSITA is assumed to have knowledge of all prior art references. I also understand that when considering the obviousness of a patent claim, one should consider whether a teaching, suggestion, or motivation to combine the references exists so as to avoid impermissibly applying hindsight when considering the prior art. I understand this test should not be rigidly applied, but that the test can be important to avoiding such hindsight.

38. I understand that a POSITA can combine various prior art references based on the teachings of those prior art references, the general knowledge present in the art, or common sense. I understand that a motivation to combine the references may be implicit in the prior art, and there is no requirement that there be an actual or explicit teaching to combine two references. Thus, one may take into account the inferences and creative steps that a POSITA would employ to combine the known elements in the prior art in the manner claimed by the patent-at-issue. I understand that one should avoid “hindsight bias” and ex post reasoning in performing an obviousness analysis. This does not mean, however, that a POSITA, for purposes of the obviousness inquiry, does not have recourse to common sense.

39. I understand that the following are examples of approaches and rationales that may be considered in determining whether a prior art reference could have been combined with other prior art or with other information within the knowledge of a POSITA:

- Combining prior art elements according to known methods to yield predictable results;
- Simple substitution of one known element for another to obtain predictable results;
- Use of known techniques to improve similar devices, methods, or products in the same way;
- Applying a known technique to a known device, method, or product ready for improvement to yield predictable results;
- That it was “obvious to try”—i.e., that it was one choice from a finite number of identified, predictable solutions, with a reasonable expectation of success;
- Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; and

- Some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify the prior art reference or to combine multiple prior art reference teachings to arrive at the claimed invention.

40. I also understand that obviousness is a legal conclusion based on the underlying factual issues of the scope and content of the prior art, the differences between the claimed invention and the prior art, the level of ordinary skill in the art, and any objective indicia of non-obviousness. For that reason, I am not rendering a legal opinion on the ultimate legal question of obviousness. Rather, my testimony addresses the underlying facts and factual analysis that would support a legal conclusion of obviousness, and when I use the term obvious, I am referring to the perspective of a POSITA at the time of invention.

VI. SUMMARY OF OPINIONS

41. For the reasons I discuss below, it is my opinion that claims 1-11 of the '321 patent are disclosed in the prior art, and are rendered obvious by the prior art.

VII. OVERVIEW OF THE '321 PATENT

A. Technology Overview

42. The '321 patent is generally concerned with encoding and decoding video sequences comprising independent sequences of image frames.¹ (EX1001, Abstract.) As its title “Grouping of Image Frames in Video Coding” suggests, a key concept discussed in the '321 patent is the grouping of these image frames. (EX1001.) The

¹ The challenged claims of the '321 patent use the term “image frames,” so I follow that convention here. The '321 patent uses the terms “image frame,” “frame,” “video frame,” and “picture” interchangeably. (See EX1001, 10:25-30 (“If there are temporally overlapping image frames on the same layer, the frames are interpreted as alternative presentations of the same image content and therefore any mode of image presentation can be used.”), 8:26-29 (“FIG. 2 further shows a next I-frame 202 of the INTRA layer, the frame thus being also an independently determined video frame that has been determined without using motion compensated temporal prediction.”), 4:20-23 (“An indication of at least one image frame is encoded into the video sequence, which indicated image frame is the first picture, in decoding order, of the independent sequence.”).) These terms are all ways to refer to a single still image. (See EX1001, 2:55-58 (explaining that “video files ... comprise a great number of still image frames, which are displayed rapidly in succession ... to create an impression of a moving image”).)

concept of grouping image frames into displayable sequences has existed since at least 1993, when International Standard ISO/IEC 11172-2 (“MPEG-1”) was released. That standard used the term “group of pictures” or “GOP” to refer to a group of image frames. (EX1005, 23² (defining group of pictures), 71 (explaining that “[a] picture corresponds to a single frame of motion video”).) The ’321 patent uses these terms in the same way. (See, e.g., EX1001, 2:34-40.)

43. A specific type of GOP discussed in the ’321 patent is the “independently decodable [GOP]” or “independent sequence.” (EX1001, 11:26-30, Abstract.) As its name suggests, an independently decodable GOP is one containing pictures that can be decoded independently of information outside of the GOP. Not all GOPs are independently decodable. (See EX1001, 3:52-56.) This is because image frames may depend on information from other frames, either within the same GOP or outside of it. This dependency arises from the use of a compression technique called “temporal prediction,” which takes advantage of “temporal redundancy” across image frames.

44. Temporal redundancy concerns the similarities between image frames. (See EX1001, 2:1-3 (“[T]emporal redundancy refers to the changes taking place in specific image objects in subsequent image frames.”).) Because image frames in a

² Page numbers for non-patent references refer to the page numbers added by Petitioner (and correspond to the pdf page number), unless otherwise stated.

video sequence are typically captured in quick succession, many aspects of the scene being captured do not change, or change in predictable ways from one image frame to the next. These aspects are considered to be temporally redundant.

45. One way to take advantage of temporal redundancies is to encode only the changes that occur between image frames. This approach is called temporal prediction because information about one frame is used to predict information about another frame from a different time. Temporal prediction allows image frames to be encoded using less data because encoding does not have to include redundant information. Instead, the encoded information can contain only some information about the depicted scene and refer to another image frame for any missing information.

46. A specific type of temporal prediction mentioned in the '321 patent is “motion-compensated temporal prediction.” (*See* EX1001, 2:18-24.) Motion compensated-temporal prediction is accomplished by predicting the location of an object in a given image frame based on its location in other image frames and information about its movement. (*See* EX1001, 2:16-23 (describing “motion-compensated temporal prediction” as follows: “the contents of some (typically most) of the image frames in a video sequence are predicted from other frames in the sequence by tracking changes in specific objects or areas in successive image frames”).)

47. When an image frame is encoded using temporal prediction, additional information outside of the encoded image frame is needed to decode it, namely, the

reference image frame on which it depends. (*See* EX1001, 2:31-33 (“If a frame is lost, frames dependent on it can no longer be correctly decoded.”).)

48. Image frames are given “types” based on their dependence on other image frames. (*See, e.g.*, EX1001, 3:50-52 (“Further, the reference picture selection can be generalized to cover all temporally predicted frame types, including B-frames.”).) The ’321 patent discusses three of these types. “Intra-frames” or “I-frames” do not depend on information from other image frames. (EX1001, 2:23-29.) “Inter-frames,” or “P-frames,” depend on information from previous I- or P-frames. (EX1001, 2:27-31.) “Bi-directionally predicted frames,” or “B-frames,” depend on both a preceding and succeeding reference frame. (EX1001, 2:43-48.)

49. As mentioned above, an independently decodable GOP or independent sequence is one that can be decoded without information from image frames in another GOP. In other words, an independently decodable GOP is one in which any image frames encoded using temporal prediction depend only on reference frames within that same independently decodable GOP.

50. Independently decodable GOPs existed long before the alleged invention of the ’321 patent. As explained in MPEG-1 at least as early as 1993, “[t]he original concept of a group of pictures was a set of pictures that could be coded and displayed independently of any other group.” (EX1005, 85.) To this end, MPEG-1 described ways to encode groups of pictures to ensure that they were independently decodable

and still compliant with the MPEG-1 standard. (EX1005, 85 (“One way to do this is to omit any B-pictures preceding the first I-picture. Another way is to allow such B-pictures, but to code them using only backward motion compensation.”).)

B. The Challenged Claims

51. Petitioners’ Petition challenges (and in this Declaration I address) claims 1-11 of the ’321 patent. Claims 1-7 are encoder-side claims and claims 8-11 are decoder-side claims. Claims 1, 4, 7, 8, 10, and 11 are independent.

52. The Challenged Claims are directed to the grouping and numbering image frames. According to the ’321 patent, this grouping and numbering helps resolve “[a] problem in the numbering of image frames [that] arises when [a] user wishes to start browsing a video file in the middle of a video sequence.” (EX1001, 11:11-13.) The ’321 patent describes the problem as follows: “When the browsing of a video file is resumed from a random point after previous browsing, discontinuity typically occurs in the image numbering,” which “[t]he decoder typically interprets ... as [an] unintentional loss of image frames.” (EX1001, 11:20-25.)

53. The ’321 patent states that this problem can be “avoided in the decoder by defining an initiation image in an independently decodable Group of Pictures GOP that is activated at the random point of the video file, and the number of the initiation image is set at zero.” (EX1001, 11:26-30.) In other words, “when activated at a random point, the decoder preferably sets the identifier of the first image frame, preferably an I-frame, of the independently decodable sub-sequence at zero.” (EX1001, 11:37-40.) This

allows the decoder to “find the correct image frame that initiates the subsequence from the video sequence of image frames.” (EX1001, 11:46-49.)

VIII. CLAIM CONSTRUCTION

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

IX. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Kim Renders Obvious Claims 1, 3-4, 6-8, 10, and 11

1. Kim

55. Kim discloses “a device and method for recording and reproducing video data on and from an optical disk using a Time Lapse recording/reproduction process.” (EX1006, 1:10-12.) Kim describes a “recording method” that involves digitizing and compressing image frames from one or more cameras to generate a bit stream as shown in Fig. 5A. (EX1006, 3:33-5:7.) Kim also describes a “reproducing method” that involves decoding a bitstream of compressed video data in response to a user’s request and outputting an analog image signal. (EX1006, 5:8-17, 5:29-34, 5:55-6:8.) Kim’s “recording method” involves encoding video and Kim’s “reproduction method” involves decoding video. (EX1006, 1:8-12, 2:6-9, 3:27-29, 3:49-65, 5:29-35, 6:9-22, Figs. 3 (*e.g.*, step S03), Fig. 4 (*e.g.*, step S30, step S38).)

56. Kim describes a bitstream with pictures grouped into a “logical structure” based on their dependencies. Kim depicts this logical structure in Figure 5A:

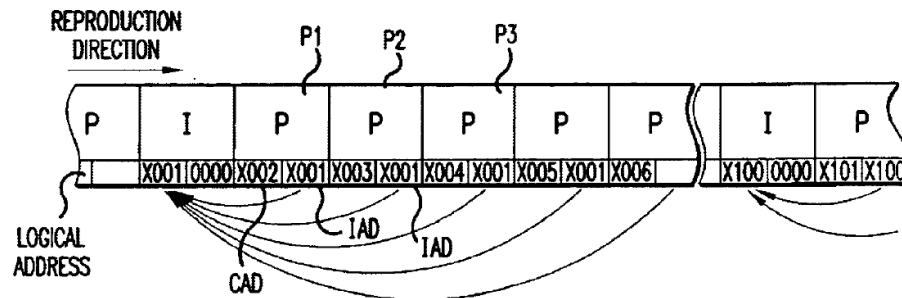


FIG.5A

(EX1006, Fig. 5A.) The logical structure comprises an I-picture followed by P-pictures that depend on it. (EX1006, Fig. 5A, 4:46-52.) Kim’s I- and P-pictures have similar characteristics to I- and P-pictures in various video coding standards including MPEG. (See EX1006, 4:15-19, 6:55-58.) Specifically, in Kim, each I-picture is an “intra-picture” and therefore “carries the entire image information of a frame.”

(EX1006, 3:60-61.) Kim’s I-pictures therefore do not depend on other pictures to be decoded and output as an analog image signal. In addition, in Kim, each P-picture is a “predictive-picture” (EX1006, 2:22) that is encoded based on “the moving vector and the position value of the corresponding (or previous) I-frame” (EX1006, 4:28-33).

Kim’s P-pictures therefore require the data from their corresponding I-picture moving vectors to be restored as full screen images. (EX1006, 5:55-59.)

57. Kim also discloses that the bitstream is comprised of data structures that associate each picture with address fields used to identify the picture and its corresponding reference picture. Kim depicts these data structures in Figure 6:

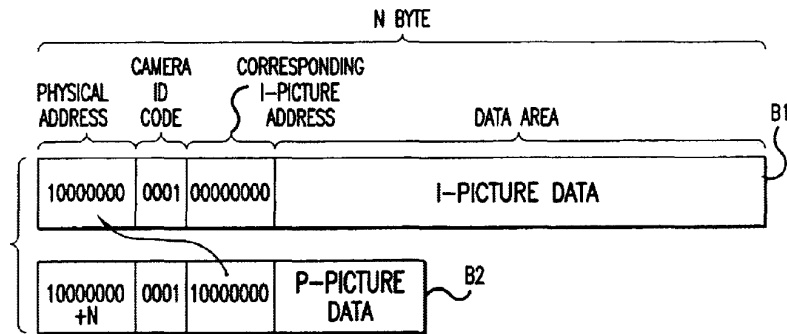


FIG.6

(EX1006, Fig. 6.)

58. Kim explains that each P-picture has a current address (“CAD”) that indicates the current address of the P-picture, and an I-picture reference address (“IAD”) that indicates the address of the corresponding (or previous) I-frame. (EX1006, 4:46-52.) I-pictures also have a CAD that indicates the picture’s current address (*see* EX1006, 4:65-5:1, Figs. 5A, 6 (“physical address”)), but the I-picture’s IAD is zero (*see* EX1006, Figs. 5A (showing “0000”), 6 (showing “00000000” for the “corresponding I-Picture address”)). Figure 5A below is annotated to highlight the IAD values in red and CAD values in green.

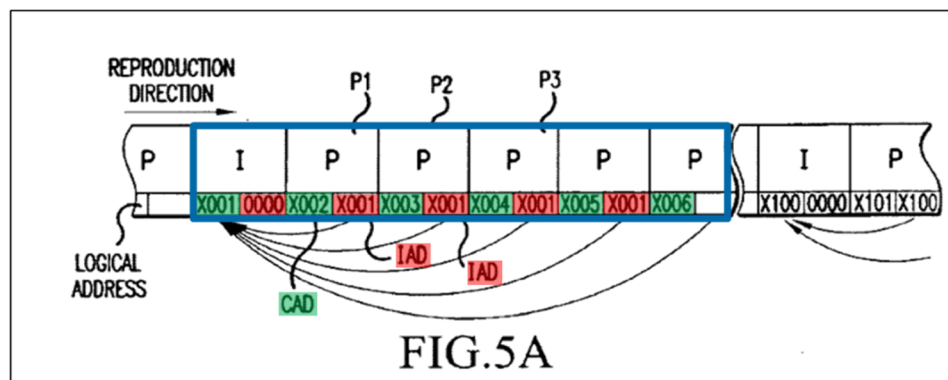
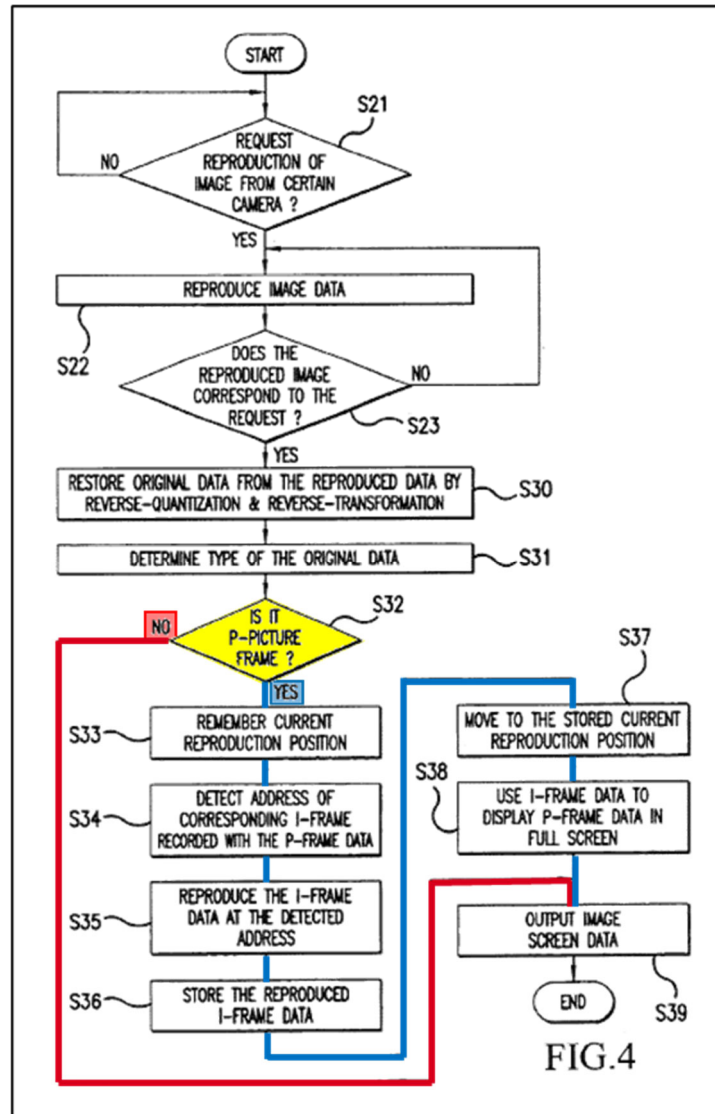


FIG.5A

(EX1006, Fig. 5A (annotated).)

59. As shown below in Kim's Figure 4, Kim sets forth two paths for decoding the disclosed bitstream depending on whether the user requests image data that corresponds to an I-picture or a P-picture. When the request corresponds to an I-picture, the picture is decoded and output as an image signal. (EX1006, 5:66-6:3, Fig. 4 ("no" path annotated in red).) When the request is directed to a P-picture, the system "detects the address of the corresponding I-picture frame ... and reproduces based on the detected address [of that] I-picture." (EX1006, 5:38-48; Fig. 4 ("yes" path annotated in blue).)



(EX1006, Fig. 4 (annotated).)

60. In my opinion, it would have been obvious to a POSITA for Kim's time lapse video frames to be displayed as a continuous sequence beginning with the picture selected by the user.

61. Time lapse video products supported continuous playback of the time-lapse video sequences. For example, EX1013 is a 1997 manual for a Sony Time Lapse

VCR (SVT-S3050P).³ The first page includes a simple explanation of Time Lapse Video and notes that playing back time lapse video increases playback speed “so that the slow changes that were recorded can be viewed at a much faster speed.”

What is Time Lapse Video?

- A normal VCR records 50 fields per second, while this time lapse VCR records in a range from 0.88 to 10 fields per second. Through reduction of the number of fields recorded per second, this VCR is capable of recording for extended periods of time on a standard video tape (up to a maximum of 168 hours).
- As a result, this VCR is extremely useful for long-term monitoring or recording events that take place slowly (such as the blooming of a flower).
- If a tape that has been recorded in time lapse mode is played back in the normal three-hour mode, the speed of the tape will be increased 5 to 57 times, depending on the speed the original recording was made at, so that the slow changes that were recorded can be viewed at a much faster speed.

EX1013 at 2; *see also id.* at 28-29 (describing adjustment of playback speed using REC/PLAY SPEED-/+ button and describing available playback speeds for different recording speeds). A GYYR products application guide from 1998 similarly explains that when time lapse recordings are played back at normal tape speed, “the playback

³ This document is available online at the following URL: <https://pro.sony/s3/cms-static-content/operation-manual/3193023022.pdf>

presents events at a speed faster than real time.” EX1014 at 3; *see also* EX10** at 12 (listing 13 different “Record/Playback Speeds” for the TLC2100 time lapse VCR).⁴

62. In my opinion, the use of time lapse video for surveillance and nature photography was well-known by the time of the alleged invention.⁵ For surveillance applications, it would have been obvious a POSITA to enable users to view time lapse sequences to identify key portions (e.g., when property was damaged or when a break-in occurred). It would have been impractical to require users to select video frames one by one that were captured over an extended period. Similarly, time lapse video sequences showing progression of day into night, clouds moving across the sky, and flowers opening up have long been used in TV programs and movies to show how things change over an extended period or event just to convey the passing of time.

2. Claim 1

- a. **1[pre]: “A method for encoding a video sequence comprising an independent sequence of image frames, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence, the method comprising:”**

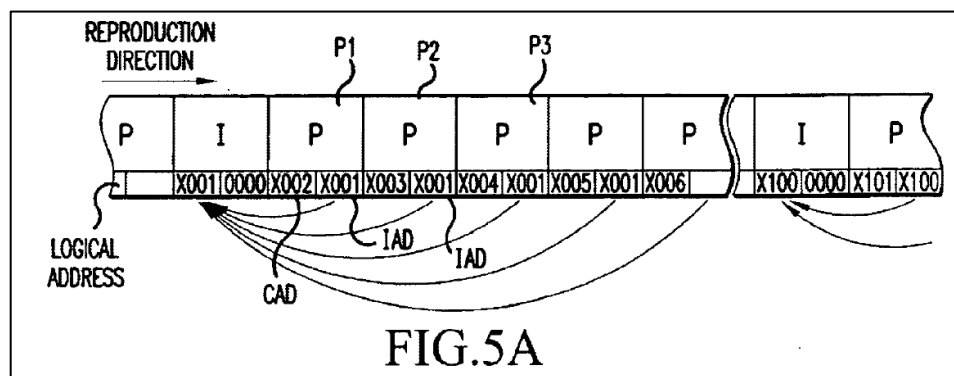
63. In my opinion, Kim teaches the preamble of claim 1.

⁴ An archived version of this document from 1998 is available at: <https://web.archive.org/web/19980628195831/http://www.gyyr.com:80/pdf/appguide.pdf>.

⁵ *See, e.g.*, EX1014 at 59 (“Who should use time-lapse recording?”).

64. Kim discloses encoding video sequences comprising image frames. For example, Kim discloses “an encoder 110 for generating I-picture data or P-picture data in frame units” (EX1006, 3:3-6.)

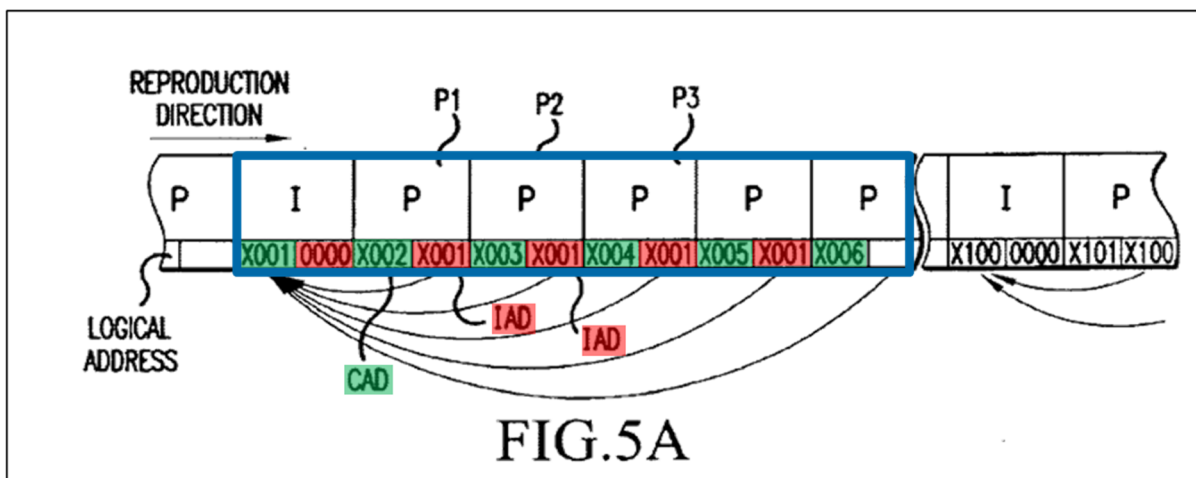
65. Kim also discloses the claimed “independent sequence” Kim’s Figure 5A (reproduced below) shows an example of a logical data structure of a digital data stream. (EX1006, 2:48-51.)



(EX1006, Fig. 5A.)

66. As can be seen, Kim’s bitstream includes intra-pictures (I-pictures) and predictive-pictures (P-pictures). (EX1006, Fig. 5A, 2:17-27.) Kim’s I-pictures are coded based only on information within themselves. (EX1006, 3:60-61 (“The I-picture frame (intra-frame) data carries the entire image information of a frame.”).) The P-pictures (also referred to in Kim as “P-frames”) are coded with “the moving vector and the position value of the corresponding (or previous) I-frame.” (EX1006, 4:15-27.) In my opinion, Kim’s grouping of an I-picture followed by P-pictures that refer to it teaches the recited “independent sequence.”

67. Kim discloses an independent sequence in which every P-picture following an I-picture depends on that I-picture for motion prediction. (EX1006, Fig. 5A, 4:46-52.) Kim explains that “each P-picture frame...has a current address CAD indicating the current address of the P-picture, and an I-picture reference address IAD indicating the address of the corresponding (or previous) I-frame.” (EX1006, 4:46-52.) Thus, each P-picture’s IAD identifies the I-picture on which the P-picture depends for motion compensation. This is further shown in annotated Figure 5A below, where every P-picture’s IAD identifies the address of the preceding I-picture and shows these dependencies as arrows pointing from each P-picture back to the preceding I-picture. (EX1006, Fig. 5A (IAD highlighted in red; CAD highlighted in green; independent sequence outlined by blue box).) Thus, every P-picture in Kim’s bitstream depends on the preceding I-picture and only on that preceding I-picture.



(EX1006, Fig. 5A (annotated).)

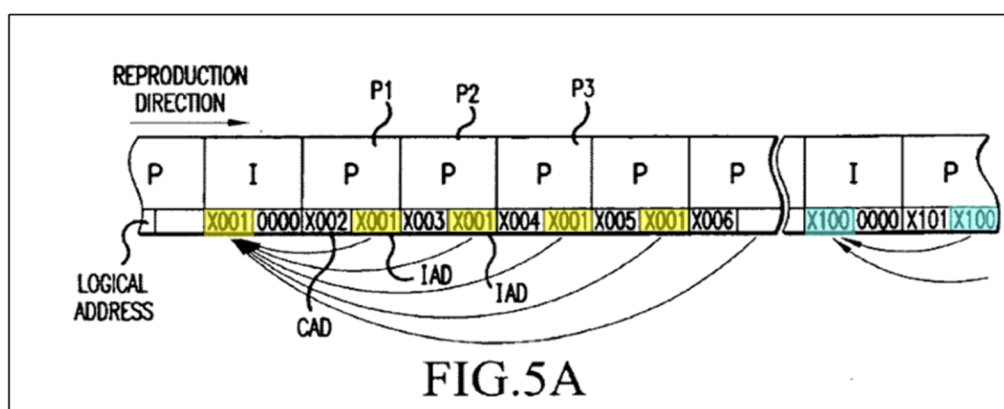
68. Because every P-picture following the I-picture depends only on that I-picture for motion prediction in the sequence shown in Figure 5A, all motion-compensated temporal prediction references of the sequence shown in Figure 5A refer only to image frames within the sequence. Accordingly, Kim discloses or suggests a group of frames where every P-picture depends only on the preceding I-picture for motion prediction (“an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within the independent sequence”).

b. 1[a]: “encoding into the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence;”

69. In my opinion, Kim teaches this limitation.

70. First, Kim teaches the recited “indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence.” Specifically, as shown in Kim’s Figure 5A, an IAD of 0000 or zero is “an indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence.” Kim discloses that each image frame has a CAD and an IAD. (EX1006 4:46-52, 4:59-5:7, Figs. 5A, 6.) Kim’s Figure 5A shows that each I-picture has an IAD of 0000. (EX1006, Fig. 5A; EX1006, 4:59-5:1 (explaining that for I-pictures “the address of the corresponding I-picture frame [(i.e., the IAD)]...is *zero* or absent”).) The IAD for each P-picture, on the other hand, is the CAD (current address) of the preceding I-picture. (EX1006, 4:48-52.)

71. Because each I-picture starts a new independent sequence, the IAD of 0000 or zero in that picture indicates that it starts an independent sequence (i.e., “is the first image frame, in decoding order, of an independent sequence”). Annotated Figure 5A below shows the CAD for the I-picture and IAD for the P-pictures of the first independent sequence highlighted in yellow and shows the CAD for the I-picture and IAD for the P-pictures of the second independent sequence highlighted in light blue.



(EX1006, Fig. 5A (annotated).)

72. In my opinion, a POSITA would have understood that Kim’s encoder encodes the IAD of the image frames. Kim’s Figure 5A provides an “example[] of a logical data structure of a digital data stream” (EX1006, 2:48-51), and “FIG. 6 shows an example of a data structure of the bit stream shown in FIG[[]]. 5A” (EX1006, 4:59-61). Kim then explains the process for encoding this bitstream. (EX1006, 3:33-4:45, Fig. 3.) A POSITA would have understood that encoding the bitstream in Figure 5A would include encoding the components of that bitstream, including the IAD values for each picture. Additionally, Kim discloses that during the encoding process for each P-

frame, “the position value of the corresponding (or previous) I-frame ... is added to the corresponding compressed P-frame output from the encoder 110.” (EX1006, 4:27-33; Fig. 5A.)

73. Accordingly, Kim teaches or suggests 1[a] by teaching the encoding of the IAD value of 0000 for each I-picture as shown, for example, in Figure 5A.

74. Taken together, in my opinion, Kim’s disclosure of encoding the IAD value of 0000 for each I-picture and Kim’s disclosure of grouping an I-picture followed by P-pictures that depend on that corresponding I-picture teach or suggest “encoding into the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence.”

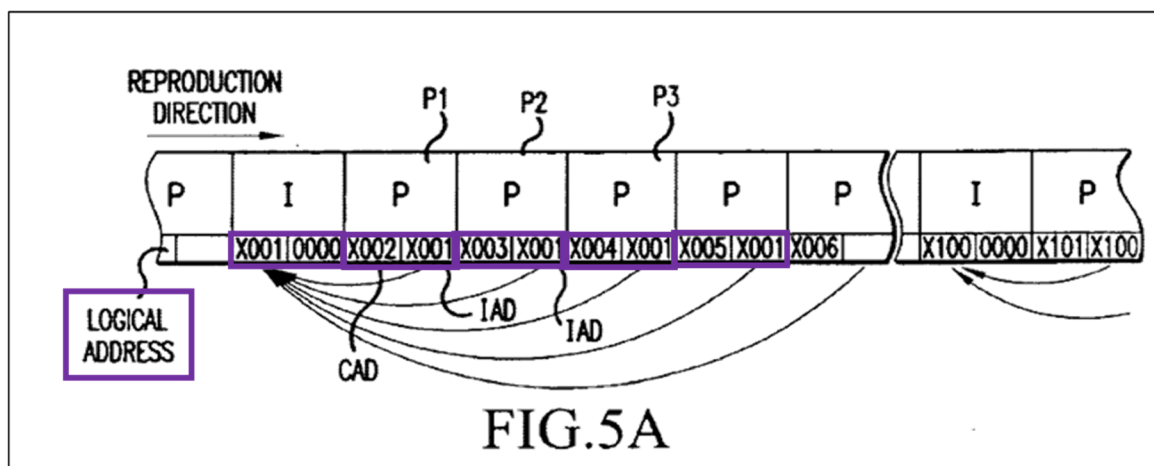
c. 1[b]: “encoding identifier values for the image frames according to a numbering scheme; and”

75. In my opinion, Kim teaches this limitation.

76. Kim’s “logical addresses” are “identifier values for the image frames” as claimed, and they are encoded “according to a numbering scheme.”

77. Kim teaches or suggests that pictures have “logical addresses” assigned according to a numbering scheme. As shown in Kim’s Figure 5A, each picture has a “logical address” comprised of the picture’s IAD and CAD. (EX1006, Fig.5A (annotated below with logical addresses outlined in purple).) For I-pictures and P-pictures, the CAD component of the logical address provides the current address of a picture. (EX1006, 4:48-52.) In Fig. 5A, the CAD increments by 1 for each picture in

reproduction order. (EX1006, Fig. 5A (showing that the first I-picture has a CAD of 0001, that the following P-picture has a CAD of 0002, and that CAD continues to increment by 1 for each subsequent P-picture in the group).) For I-pictures, the IAD component is zero. (EX1006, Fig. 5A.) And for P-pictures, the IAD references the CAD of the corresponding I-picture. (EX1006, 4:48-52.) Accordingly, the pictures (*i.e.*, image frames) in Kim’s disclosed bitstream have logical addresses that follow a numbering scheme.



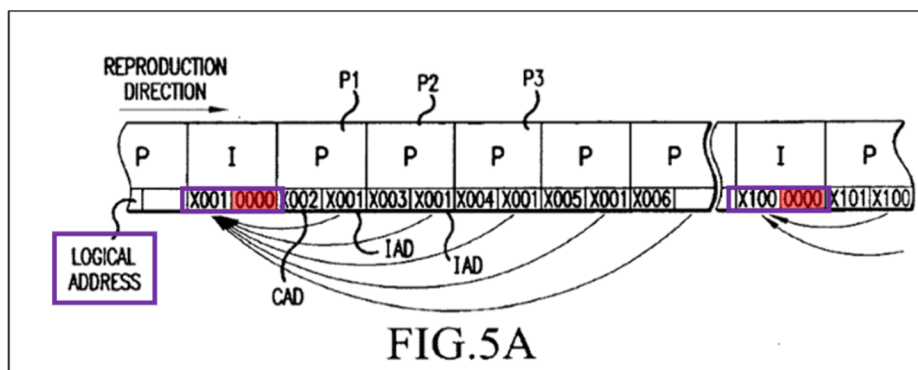
(EX1006, Fig. 5A (annotated).)

78. Moreover, given Kim’s disclosure that the addresses for each image are part of the bitstream as shown in Fig. 5A, a POSITA would have understood that the address numbers are encoded. Additionally, Kim discloses that during the encoding process for each P-frame, “the position value of the corresponding (or previous) I-frame ... is added to the corresponding compressed P-frame output from the encoder 110.” (EX1006, 4:27-33; Fig. 5A.)

d. 1[c]: “resetting the identifier value for the indicated first image frame of the independent sequence.”

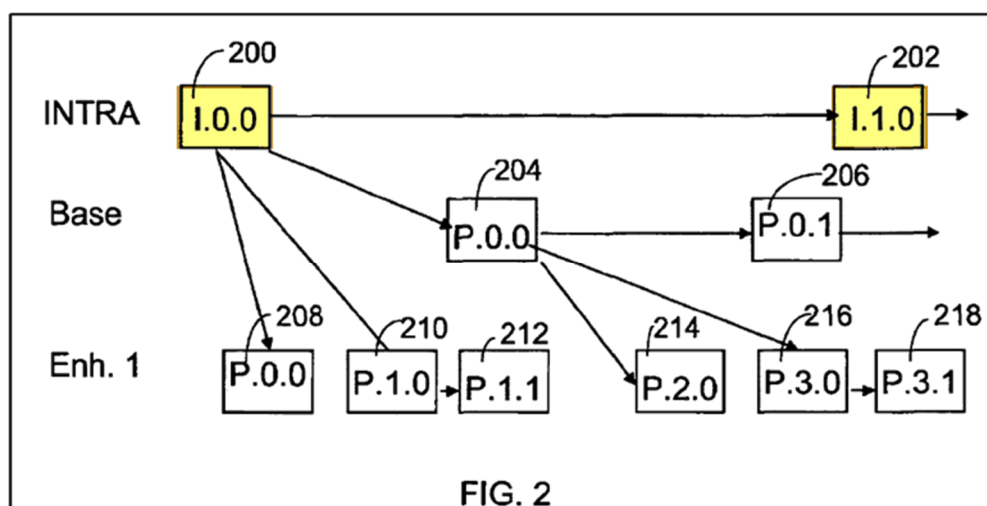
79. In my opinion, Kim teaches this limitation.

80. Specifically, Kim’s Figure 5A shows that the IAD component of each logical address is reset to zero for each I-picture. (EX1006, Fig. 5A (annotated below to show IAD component of each I-picture’s logical address in red and the whole logical address outlined in purple).) Because, as I discussed, each I-picture is the first image frame of an independent sequence, Kim discloses or suggests that the IAD component of each logical address (“identifier value”) is reset for the indicated first image frame of the independent sequence.



(EX1006, Fig.5A.) Because each P-picture has an IAD that references the prior I-picture, the value of the IAD for the P-picture preceding a given I-picture would reference the previous I-picture. A POSITA would have understood from Kim, however, that when assigning the IAD to the next I-picture, however, the encoder would reset the IAD to zero because unlike the previous P-picture, the I-picture begins a new independent sequence and does not depend on the previous I-picture.

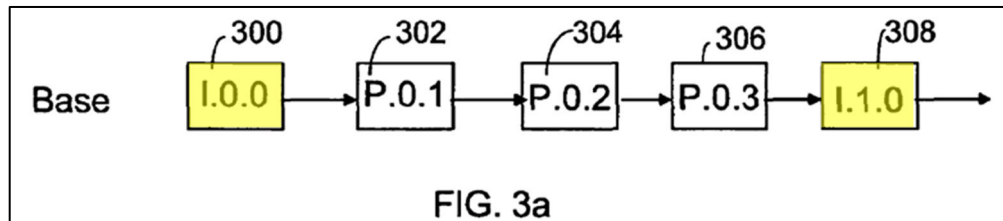
81. To the extent Patent Owner argues that Kim's disclosure of resetting a *component* of an identifier value does not teach or suggest this limitation, I would disagree because resetting a component of the identifier value of an indicated first image frame is consistent with the '321 patent's own teachings. In fact, the only two disclosures I see in the '321 patent addressing more than one independent sequence, Figures 2 and 3a, show only a component (*i.e.*, the image number) of the identifier being reset.



(EX1001, Fig.2 (annotated).)

82. In Figure 2 of the '321 patent, I-frames 200 and 202 reflect the first image frames of independent sequences. (EX1001, Fig.2.) I-frame 200 has a type I and a subsequence identifier and image number of (0.0). (EX1001, 8:22-25.) I-frame 202 has a type I and subsequence identifier and image number of 1.0. (EX1001, 8:36-40.) The '321 patent explains that the identifier for I-frame 202 is therefore I.1.0, a combination of the type, subsequence identifier, and image number. (EX1001, 8:41.)

The '321 patent further explains that the next I-frame would have an identifier of I.2.0. Thus, only a component of the '321 patent's disclosed identifiers, the image number, resets. The subsequence identifier component does not reset.



83. Figure 3a of the '321 patent likewise shows that, within the meaning of the '321 patent, “resetting an identifier value” includes resetting a component of an identifier value. The '321 patent states that “[t]he video sequence comprises a first image frame 300 which is an I-frame (I.0.0) and which thus initiates a first subsequence.” (EX1001, 12:24-26.) According to the '321 patent, “a second subsequence” is initiated by “an I-frame (I.1.0) coded therein, i.e. an I-frame 308.” (EX1001, 12:31-34.) Thus, again, only a component of the identifier value is reset when a new subsequence is initiated. This confirms that Kim teaches “resetting an identifier value” in the same way as the '321 patent discloses.

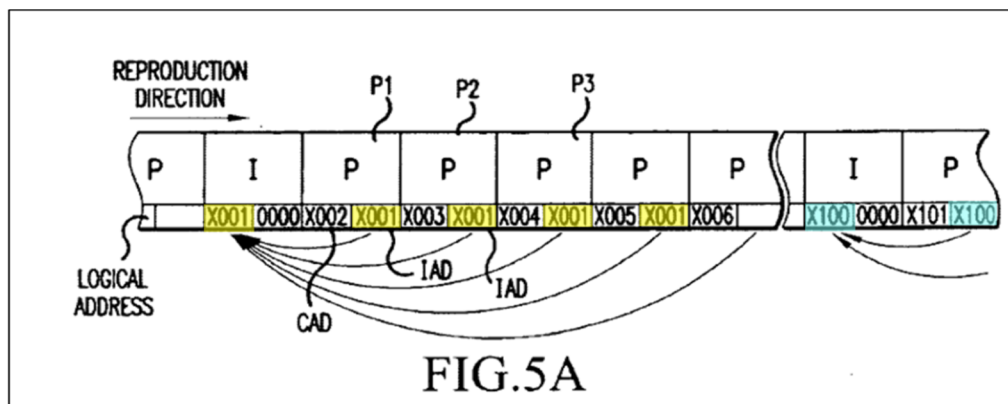
84. Accordingly, Kim teaches or suggests resetting the identifier value for the indicated first image frame of the independent sequence through its disclosure that the IAD portion of a picture's logical address is reset for the first frame of an independent sequence.

85. For the reasons I have discussed above, it is my opinion that claim 1 is unpatentable as obvious over Kim.

3. Claim 3: “A method according to claim 1, further comprising: encoding into the video sequence an identifier value for the independent sequence.”

86. In my opinion, Kim teaches claim 3.

87. As I discussed above, Kim discloses or suggests that each I-picture begins a new independent sequence. Each I-picture also has a unique CAD. For example, in Fig. 5A, the first I-picture is numbered X001 and the subsequent I picture-is numbered X100.



(EX1006, Fig. 5A (annotated).)

88. As also shown in the figure, each P frame explicitly references its corresponding I frame in the IAD portion of its own logical address. Thus every frame in every sequence includes an explicit reference to the CAD of the I frame that begins the sequence. In my opinion, a POSITA would have understood that these explicit references to the CAD of the I frame that marks the start of the sequence are “identifier

values for the independent sequence” as claimed. Furthermore, the fact that these values are included in the bitstream shows that they are “encoded.” (See EX1006, 4:27-33 (“the position value of the corresponding (or previous) I-frame ... is added to the corresponding compressed P-frame output from the encoder 110”).)

4. Claim 4

- a. 4[pre]: “A video encoder comprising at least one processor and at least one memory including computer program code, the at least one memory and the computer program code configured to, with the at least one processor, cause the video encoder to encode a video sequence comprising an independent sequence ... by causing the video encoder to at least:”

89. In my opinion, Kim teaches the preamble of claim 4.

90. Kim discloses a Time Lapse recording/reproduction device that includes “an encoder 110 for generating I-picture data or P-picture data in frame units by using transformation and quantization of image data output from the frame memory 100”; a recording-digital signal processor (DSP) 120 for extracting moving vectors from the image data output from the frame memory 100 and for generating P-picture data of the compressed image data”; and “a controller 150 for controlling the above-described components based on the commands from the user.” (EX1006, 3:3-27, Fig. 2.)

91. A POSITA would have understood that the disclosed controller and corresponding encoder and recording DSP comprise “at least one processor and at least one memory including computer program code configured to, with the at least one processor, cause the video encoders to encode a compressed video sequence.” In my

opinion, a POSITA at the time of the alleged invention would have been familiar with DSPs as specialized computer processors that execute stored programs of computer instructions, often for performing signal processing operations. (*See, e.g.*, EX1012 at 339 (describing DSPs for data compression including “programmable chips that may be downloaded with various algorithms”).) Thus, Kim’s disclosure of the DSP by itself is sufficient to teach “at least one processor and at least one memory including computer program code configured to, with the at least one processor, cause the video encoder to encode a video sequence.”

92. Additionally, Kim discloses encoding image frames in accordance with MPEG. (EX1006, 4:15-19.) By the time of the alleged invention, software-based MPEG encoders were known and commercially available. (*See, e.g.*, EX1011 at 366 (identifying software-based MPEG encoders by vendor); *id.* at 367-381; *see also, e.g.*, EX1015 (data sheet for C-Cube CLM4111 MPEG-1 Video Encoder for “mainstream consumer applications”).) By 1998, there were software-based reference encoders and decoders for MPEG-2. (EX1017.) Accordingly, it would have been obvious to a POSITA to implement an MPEG encoder with a processor, memory and computer code as claimed. Accordingly, it would have been obvious to a POSITA to similarly implement Kim’s encoder with a processor, memory and computer code as claimed.

93. Claim 4[pre] recites the same “independent sequence” as 1[pre] which is disclosed by Kim as I discussed above.

- b. 4[a]: “encode into the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence;”**

94. In my opinion, Kim teaches this limitation.

95. This limitation is almost identical to 1[a] and is obvious for the same reasons I discussed in Section IX(A)(2)(b).

- c. 4[b]: “encode identifier values for the image frames according to a numbering scheme; and”**

96. In my opinion, Kim teaches this limitation.

97. This limitation is almost identical to 1[b] and is obvious for the same reasons I discussed in Section IX(A)(2)(c).

- d. 4[c]: “reset the identifier value for the indicated first image frame of the independent sequence.”**

98. In my opinion, Kim teaches this limitation.

99. This limitation is almost identical to 1[c] and is obvious for the same reasons I discussed in Section IX(A)(2)(d).

- 5. Claim 6: “A video encoder according to claim 4, wherein the video encoder is configured to encode into the video sequence an identifier value for the independent sequence.”**

100. In my opinion, Kim teaches claim 6.

101. Claim 6 adds a nearly identical limitation to that recited by claim 3 and is obvious for the same reasons I discussed in Section IX(A)(3).

6. Claim 7

- a. 7[pre]: “A computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for encoding a video sequence comprising an independent sequence of image frames, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence, the computer program product comprising:”**

102. In my opinion, Kim teaches the preamble of claim 7.

103. As I discussed in Section IX(A)(2)(a) while addressing 1[pre], Kim teaches “encoding a compressed video sequence comprising an independent sequence of image frames” Kim also teaches the additional elements of 7[pre].

104. Kim discloses a Time Lapse recording/reproduction device that includes “an encoder 110 for generating I-picture data or P-picture data in frame units by using transformation and quantization of image data output from the frame memory 100”; “a recording-digital signal processor (DSP) 120 for extracting moving vectors from the image data output from the frame memory 100 and for generating P-picture data of the compressed image data”; and “a controller 150 for controlling the above-described components based on the commands from the user.” (EX1006, 3:3-27, Fig. 2.)

105. In my opinion, a POSITA would have understood these components to be “data processing devices,” given Kim’s disclosure that they all process data in the form of image data or commands.

106. Moreover, a POSITA would have found it obvious to implement Kim's encoder 110, recording DSP 120, and controller 150 as "[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for encoding a video sequence" As I discussed above, a DSP is a specialized type of computer processor that executes a stored program of computer instructions, often for performing signal processing operations. A POSITA would have understood that Kim's recording DSP 120 necessarily performs its portion of the encoding process by executing a "computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for encoding a video sequence" Likewise, a POSITA would have found it obvious in my opinion to implement the remaining video encoding functions performed by Kim's encoder 110 and controller 150 as a stored computer program running on a general purpose processor that was available at the time. Finally, a POSITA would have understood that these computer program products would have been stored on a non-transitory computer readable medium so that they could be stored and then executed by the one or more processors to decode a video sequence based on the commands from the user.

107. Furthermore, Kim discloses encoding image frames in accordance with MPEG. (EX1006, 4:15-19.) As I noted above, by the time of the alleged invention, software-based MPEG encoders were known and commercially available there were software-based reference encoders and decoders for MPEG-2. It would have been

obvious to a POSITA to implement an MPEG encoder with a processor, memory and computer code as claimed. Accordingly, it would have been obvious to a POSITA in my opinion to similarly implement Kim's encoder with a processor, memory and computer code as claimed.

- b. 7[a]: "a computer program code for encoding into the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of the independent sequence;"**

108. In my opinion, Kim teaches this limitation.

109. This limitation is almost identical to 1[a] and is obvious for the same reasons I discussed in Section IX(A)(2)(b).

- c. 7[b]: "a computer code for encoding identifier values for the image frames according to a numbering scheme; and"**

110. In my opinion, Kim teaches this limitation.

111. This limitation is almost identical to 1[b] and is obvious for the same reasons I discussed in Section IX(A)(2)(c).

- d. 7[c]: "a computer code for resetting the identifier value for the indicated first image frame of the independent sequence."**

112. In my opinion, Kim teaches this limitation.

113. This limitation is almost identical to 1[c] and is obvious for the same reasons I discussed in Section IX(A)(2)(d).

7. Claim 8

114. Claim 8 is the first of the '321 patent's decoder-side claims. A POSITA would have known that decoders typically operate by inverting the steps performed during encoding. Here, the decoder-side claims of the '321 patent recite similar limitations to the encoder-side claims I discussed above.

a. 8[pre]: “A method for decoding a compressed video sequence, the method comprising”

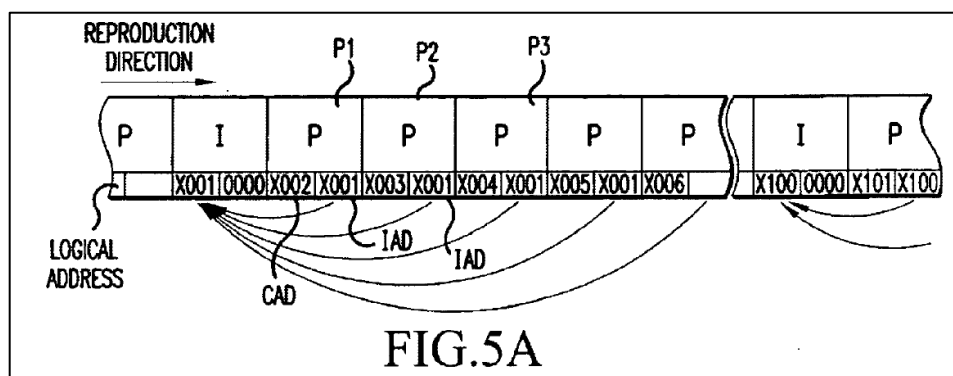
115. In my opinion, Kim teaches the preamble of claim 8.

116. Kim is directed to “reproducing video data on and from an optical disk using a Time Lapse recording/reproduction process.” (EX1006, 1:11-12.) Kim “provide[s] a recording method for compressing and intermittently recording image data on a recording medium with a maximum recording capacity.” (EX1006, 2:3-6.) Kim further “provide[s] a reproduction method for reproducing [the] compressed intermittent data from a recording medium.” (EX1006, 2:7-9.) Kim’s reproduction method includes decoding image data with a decoder. (EX1006, 5:29-31.) Accordingly, in my opinion, Kim discloses a “method for decoding a compressed video sequence” as claimed.

b. 8[a]: “decoding from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence”

117. In my opinion, Kim teaches this limitation.

118. Kim's Figure 5A (reproduced below) shows an example of a logical data structure of a digital data stream. (EX1006, 2:48-51.) In my opinion, this figure and corresponding disclosures teach the recited "indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence."

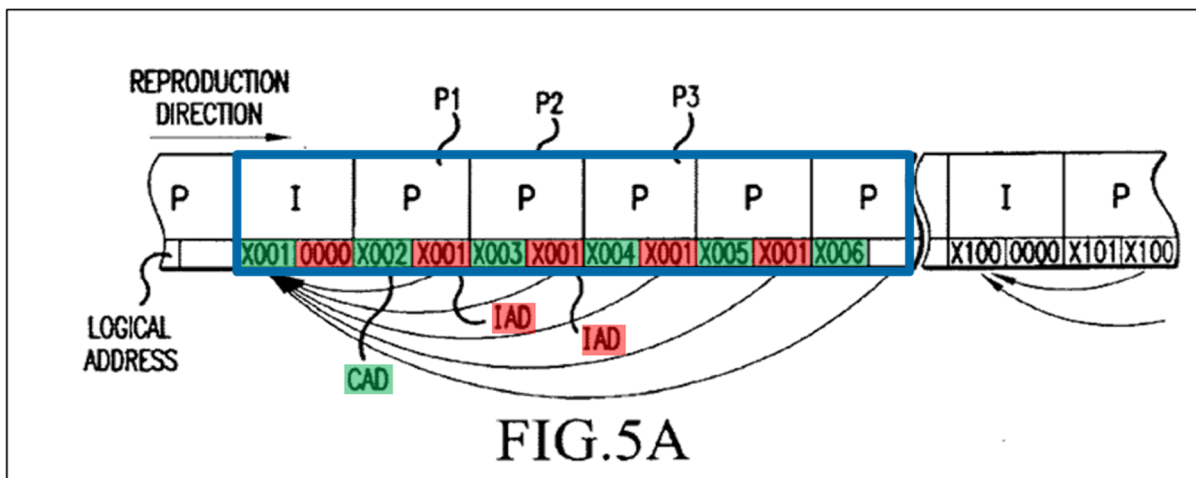


(EX1006, Fig. 5A.)

119. Kim's bitstream includes intra-pictures (I-pictures) and predictive-pictures (P-pictures). (EX1006, Fig. 5A, 2:17-27.) Kim's I-pictures are coded based only on information within themselves. (EX1006, 3:60-61 ("The I-picture frame (intra-frame) data carries the entire image information of a frame.")) The P-pictures (also referred to in Kim as "P-frames") are coded with "the moving vector and the position value of the corresponding (or previous) I-frame." (EX1006, 4:15-27.) In my opinion, Kim's grouping of an I-picture followed by P-pictures that refer to it teaches the recited "independent sequence."

120. Kim discloses an independent sequence in which every P-picture following an I-picture depends on that I-picture for motion prediction. (EX1006, Fig. 5A, 4:46-

52.) Kim explains that “each P-picture frame . . . has a current address CAD indicating the current address of the P-picture, and an I-picture reference address IAD indicating the address of the corresponding (or previous) I-frame.” (EX1006, 4:46-52.) Accordingly, Kim discloses that each P-picture’s IAD identifies the I-picture on which the P-picture depends for motion compensation. This is further shown in annotated Figure 5A below, where every P-picture’s IAD identifies the address of the preceding I-picture and shows these dependencies as arrows pointing from each P-picture back to the preceding I-picture. (EX1006, Fig. 5A (IAD highlighted in red; CAD highlighted in green; independent sequence outlined by blue box).) Thus, every P-picture in Kim’s bitstream depends on the preceding I-picture.

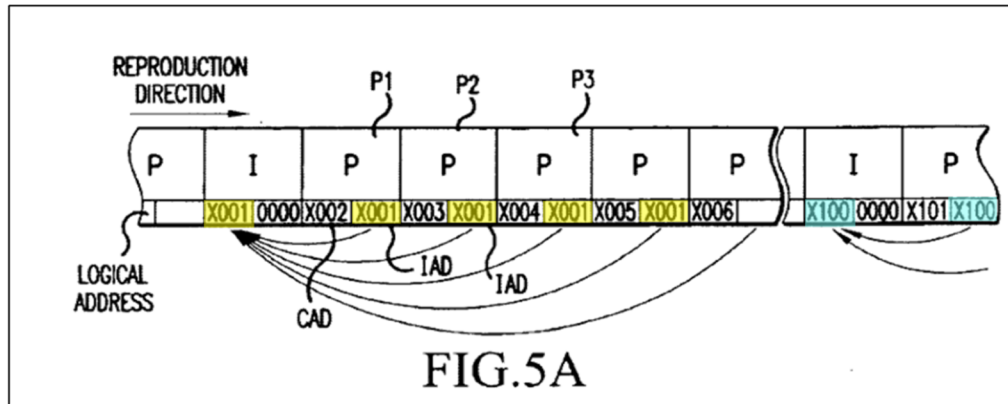


(EX1006, Fig. 5A (annotated).)

121. Because every P-picture following the I-picture depends on that I-picture for motion prediction in the sequence shown in Figure 5A, all motion-compensated temporal prediction references of the sequence shown in Figure 5A refer only to image

frames within the sequence. It is therefore my opinion that Kim teaches “an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within the independent sequence.”

122. In my opinion, Kim also teaches the recited “indication.” Specifically, as shown in Kim’s Figure 5A, an IAD of 0000 or zero is “an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence.” As explained in Kim, each image frame in the bitstream has a CAD and an IAD. (*See* EX1006, 4:46-52, 4:59-5:7, Figs. 5A, 6.) Kim’s Figure 5A shows that each I-frame has an IAD of 0000. (*See* EX1006, Fig. 5A; *see also* EX1006, 4:59-5:1 (describing “a data structure of the bit stream shown in FIG[. 5A” and explaining that for I-pictures “the address of the corresponding I-picture frame [(i.e., the IAD)] ... is zero or absent”).) The IAD for each P-picture, on the other hand, is the CAD (current address) of the preceding I-picture. (EX1006, 4:48-52.) Because each I-picture starts an independent sequence, the IAD of 0000 indicates that a given frame starts an independent sequence (i.e., “is the first image frame, in decoding order, of an independent sequence”). Annotated Figure 5A below shows the CAD for the I-picture and IAD for the P-pictures of the first independent sequence highlighted in yellow and shows the CAD for the I-picture and IAD for the P-pictures of the second independent sequence highlighted in light blue.



(EX1006, Fig. 5A (annotated).)

123. A POSITA would have understood that the decoder disclosed in Kim decodes the IAD of the image frames. Kim discloses that Figure 5A provides an “example[] of a logical data structure of a digital data stream” (EX1006, 2:48-51) and that “FIG. 6 shows an example of a data structure of the bit stream shown in FIG[. 5A” (EX1006, 4:59-61.) Kim then explains the process for decoding this bitstream. (See EX1006, 5:8-6:8, Fig. 4.) A POSITA would have understood that decoding the bitstream in Figure 5A would include decoding the components of that bitstream, including the IAD values.

124. Accordingly, Kim discloses limitation 8[a] by decoding the IAD value of 0000 for the I-frame in Kim’s Figure 5A.

125. It is therefore my opinion that, taken together, Kim’s disclosure of decoding the IAD value of 0000 for the I-picture and its disclosure of a grouping of I-pictures followed by P-pictures that depend on the corresponding I-picture for each group teach “decoding from the video sequence an indication of at least one image

frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence.”

- c. **8[b] “starting the decoding of the video sequence from said first image frame of the independent sequence, whereby the video sequence is decoded without prediction from any image frame decoded prior to said first image frame”**

126. In my opinion, Kim teaches this limitation.

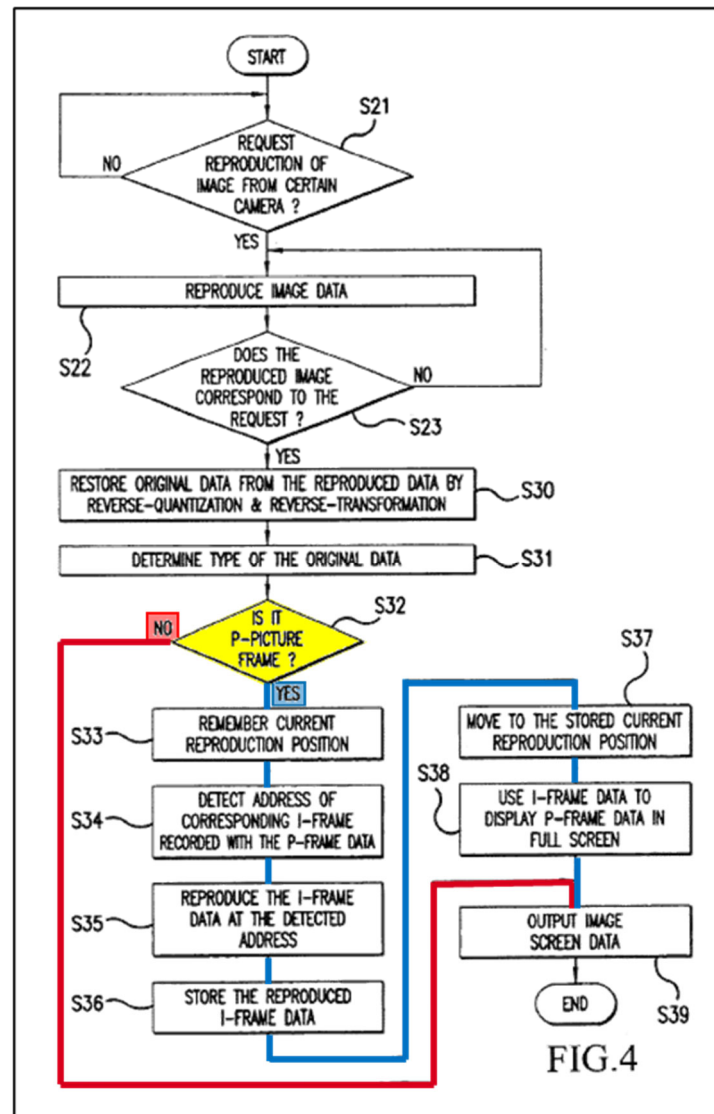
127. As discussed in Section IX(A)(7)(b) for 8[a], Kim’s disclosures of a grouping of an I-picture followed by P-pictures teaches the recited “independent sequence.” As to 8[b], Kim teaches starting the decoding of that grouping with the I-picture (“starting the decoding of the video sequence from said first image frame of the independent sequence”). Specifically, as shown in Figure 4, Kim discloses two paths for decoding an image in response to a user requesting playback. Either path discloses or suggests 8[b] in my opinion.

128. A POSITA would have understood that Kim’s Figure 4 process is applied, or would have found it obvious to apply Kim’s Figure 4 process, to each image frame in a video sequence, including the sequence in Kim’s Figure 5A. A POSITA would have been motivated to apply Kim’s Figure 4 process not only to display a single video frame but also to display Kim’s Figure 5A sequence of video frames. A POSITA would have known that video playback involves displaying a sequence of still images over time. (EX1001, 1:55-58 (“The video files in multimedia files comprise a great number

of still image frames, which are displayed rapidly in succession ... to create an impression of a moving image.”.) Further, as I discussed above in Section IX(A)(1), time lapse video products supported playback of time lapse video sequences, and a POSITA would have appreciated the practical importance of that capability to time lapse video applications.

129. Kim discloses that the flowchart in Figure 4 depicts a reproducing method for a data structure, such as the one shown in Figure 6. (EX1006, 6:8-10.) Kim further discloses that Figure 6 shows an example of a data structure of the bitstream shown in Figure 5A. (EX1006, 4:59-60.) In my opinion, a POSITA therefore would have understood that the decoding steps in Figure 4 can be repeated for each image in an independent sequence shown in Figure 5A. A POSITA would have found this combination to be no more than applying prior art elements (Kim’s Figure 4 process to Kim’s Figure 5A data structure) according to their known functions to achieve predictable results.

130. Kim's Figure 4 is reproduced below:



(EX1006, Fig. 4 (annotated).)

131. Kim's first path (red) discloses that the first image decoded and output for display is an I-picture. (See EX1006, Fig. 4 (showing "no" branch at Step S32 annotated in red).) Kim explains that "in Step S32, if the detected image data is not a P-picture frame, then the detected image data is decoded by the decoder 111 as described above, and output as an analog image signal from the D/A converter 91 in Step S39." (EX1006,

5:66-6:3.) Kim's use of the phrase "not a P-picture" means that the picture is an I-picture because Kim's sequence includes only I- and P-pictures. (*See, e.g.*, EX1006, Figs. 5A, 6.) As I explained previously, Kim's Figure 5A shows that each independent sequence begins with an I-picture. Accordingly, this path, which begins by decoding an I-picture, discloses "starting the decoding of the video sequence from said first image frame of the independent sequence."

132. The second path (blue) reflects the system's behavior when the user requests reproduction of a P-picture. (*See* EX1006, Fig. 4 (showing "yes" branch at Step S32 annotated in blue).) In the second path, "if the currently restored image data is detected to be P-picture frame data in Step S32, the controller 150 stores the current reproduction position in Step S33." (EX1006, 5:38-41.) The system then detects the address of the corresponding I-picture on which the P-picture depends for motion compensation. (EX1006, 5:41-48.) The system then decodes the I-picture. (EX1006, 5:49-52.) After the I-picture is decoded, the P-picture is restored as a full-screen image using the I-picture frame data and moving vector, and is ultimately output as a reproduction signal. (EX1006, 5:55-65.)

133. A POSITA would have understood this second path to disclose "starting the decoding of the video sequence from said first image frame of the independent sequence" because the system decodes the I-picture (the first image frame of the independent sequence) before it can produce the P-picture that depends on it. In other

words, a POSITA would have understood the P-picture to have been decoded only after the information on which it depends was available and used to decode the P-picture. This is consistent with the '321 patent's disclosure that a dependent frame "can no longer be correctly decoded" if the frame on which it depends is lost. (EX1001, 2:31-33.)

134. As I explained above, a POSITA would have found it obvious to apply Kim's Figure 4 process to each image frame in a video sequence, including the sequence in Kim's Figure 5A. But it is also worth noting in the alternative that Figure 4 itself contemplates a video sequence that includes an I-picture and a P-picture because at Step 32, if the user requests reproduction of a P-picture, the system then detects the address of the corresponding I-picture on which the P-picture depends for motion compensation and decodes that I-picture. (See EX1006, Fig. 4 (showing "yes" branch at Step S32 annotated in blue), 5:38-52.) Thus, even without applying Figure 4's process to all of the pictures of an independent sequence in Figure 5A, a POSITA would have understood that Figure 4's process applies to an independent sequence consisting of a P-picture and a corresponding I-picture. This is sufficient to teach "starting the decoding of the video sequence from said first image frame of the independent sequence."

135. Kim also discloses that the video sequence is decoded without prediction from any image frame decoded prior to said first image frame, as recited in 8[b]. As I

explained in Section IX(A)(7)(b), each independent sequence in Kim's bitstream consists of an I-frame followed by a series of P-pictures that depend on the I-frame for prediction. (See EX1006, Fig. 5A.) The I-frame that begins the sequence is intra-coded and thus does not depend on any other picture at all. (EX1006, 3:60-61 ("The I-picture frame (intra-frame) data carries the entire image information of a frame.")) And because every P-picture refers to the first I-frame in each independent sequence, no P-picture requires prediction from any image frame decoded prior to the first I-picture.

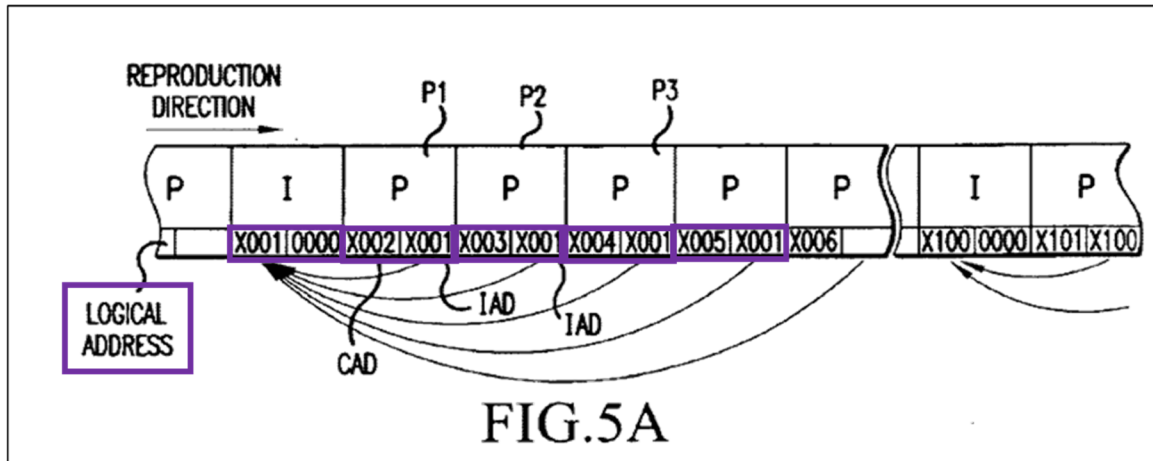
136. In view of the foregoing discussion, it is my opinion that Kim teaches "the video sequence is decoded without prediction from any image frame decoded prior to said first image frame."

137. In sum, either of the decoding paths described in Kim's Figure 4 when applied to Figure 5A's independent sequence, or alternatively the "yes" path from Step 32 in Kim's Figure 4 when applied to an independent sequence with a P-picture and corresponding I-picture, discloses "starting the decoding of the video sequence from said first image frame of the independent sequence," and under either path, Kim discloses decoding the independent sequence "without prediction from any image frame decoded prior to said first image frame."

d. 8[c] “decoding identifier values for image frames according to a numbering scheme”

138. In my opinion, Kim teaches this limitation. Specifically, Kim’s “logical addresses” are “identifier values for image frames,” and they are decoded “according to a numbering scheme.”

139. Kim teaches that pictures have “logical addresses” that are assigned according to a numbering scheme. As shown in Kim’s Figure 5A, each picture has a “logical address” comprised of the picture’s IAD and CAD. (EX1006, Fig. 5A (annotated below with logical addresses outlined in purple).) For I-pictures and P-pictures, the CAD component of the logical address provides the current address of a picture. (See EX1006, 4:48-52.) CAD increments by 1 for each picture in reproduction order. (See EX1006, Fig. 5A (showing that the first I-picture has a CAD of 0001, that the following P-picture has a CAD of 0002, and that CAD continues to increment by 1 for each subsequent P-picture in the group).) For I-pictures, the IAD component is zero. (See EX1006, Fig. 5A.) And for P-pictures, the IAD provides the CAD of the corresponding I-frame. (See EX1006, 4:48-52.) Thus, the pictures (i.e., image frames) in Kim’s disclosed bitstream have logical addresses that follow a numbering scheme.



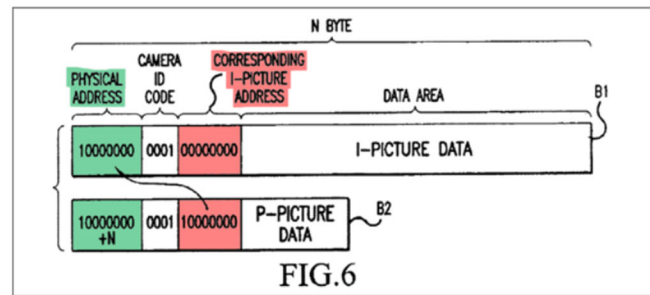
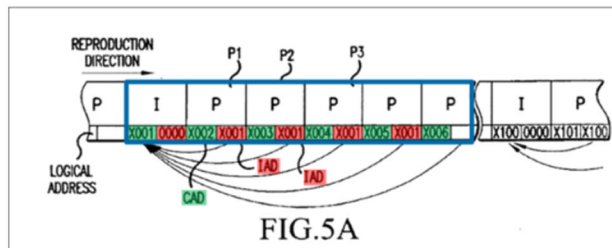
(EX1006, Fig. 5A (annotated).)

140. Kim also teaches that the logical addresses are decoded according to a numbering scheme. Specifically, Kim discloses that when a P-picture is decoded, “the decoder detects the address of the corresponding I-picture frame (e.g., data stored in the area (3) of the bit stream B2 shown in Fig. 6) ... and reproduces based on the detected address.” (EX1006, 5:41-46.) Thus, the decoder uses the logical addresses to determine the order in which pictures are decoded and reproduced. In reproducing the pictures in this order, the decoder also decodes the logical addresses. Accordingly, Kim discloses that logical addresses are decoded according to a numbering scheme.

141. In IPR2024-00691, Patent Owner tried to argue that the logical addresses in Fig. 5A would not be decoded because the description of Fig. 6 refers to a “physical address” rather than a logical address and this physical address would be decoded. I disagree. The term “physical address” appears once and only once in the entirety of Kim in describing Fig. 6. (EX1006, ** (stating that the data structure of the bit stream carries “the physical address of the bit stream (1)”). Kim is clear that Fig. 6 is showing

data structures for each picture from the bitstream shown in Figs. 5A and 5B. (EX1006, 4:59-61 (“FIG.6 shows an example of *a data structure of the bit stream shown in FIGS. 5A and 5B* to be recorded and reproduced on and from the optical disk medium.”).) A POSITA would understand Fig. 6 to be depicting the same underlying data as Figs. 5A and 5B but simply separating the data into data structures for each.

142. In my opinion, the “physical address” Kim references in describing Fig. 6 clearly correlates to the CAD shown in Fig. 5A. Consistent with Kim’s Fig. 5A and its description of that figure, Kim’s Fig. 6 shows how the I-Picture Address (IAD) for the P-picture references the “Physical Address” (or CAD) of the corresponding I-picture.



(EX1006, Fig. 6.) That the physical address for the second frame in Fig. 6 is shown as “10000000+N” would have suggested to a POSITA that the address increments by one for each of the N frames which is also consistent with the CAD values in Fig. 5A. Thus, in my opinion, a POSITA would have understood that the “physical address” Kim references in describing Fig. 6 is the CAD portion of the logical address Kim shows and describes in connection with Fig. 5A.

143. Additionally, Kim suggests the address information is determined during the encoding process *before* any data is written to disk. For example, in Step S11, “the

position value of the corresponding (or previous) I-frame ... is added to the corresponding compressed P-frame output from the encoder 110.” (EX1006, 4:29-33.) Later, in Step S13, “the MUX 130 selectively outputs the I-picture data (or P-picture data), the camera ID code, and *the address of the I-picture data received from the encoder 110, the recording DSP 120 and the controller 150*, and generates a bit stream of certain bytes for each frame [which are] transmitted to the recording/reproducing unit 140 which records them on the optical disk 1.” (EX1006, 4:38-45.) Nothing in Kim suggests the encoder has (or needs) advance knowledge of where the frames will be stored on the optical disk and uses that information to set the picture addresses.

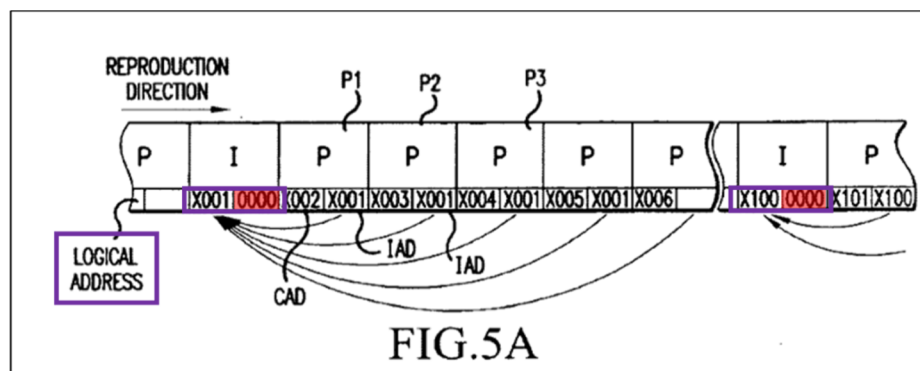
144. Additionally, in my opinion, a POSITA would have thought it strange and undesirable for a compressed video bitstream to rely on precise physical addresses tied to a particular storage medium as such an approach would cause problems when data is copied from one storage medium to another and require reencoding every time the same compressed video is stored on a different medium. Notably, well-known video standards such as MPEG, which Kim specifically mentions, did not use physical addresses tied to particular storage media.

145. Even if a POSITA interpreted Kim’s lone reference to a “physical address” to suggest the use of a specific physical address on the optical disk, a POSITA would understand that the CAD in Fig. 5A would use the same address given the consistency between Figs. 5A and 6. Patent Owner’s argument that the CAD in Fig. 5A and the

“physical address” in Fig. 6 are different is simply not supported by the reference in my opinion. To the extent the argument is based on the different meanings of the terms “logical address” and “physical address,” a POSITA would not attribute so much weight to that distinction in my view in the context of Kim’s teachings. Kim’s Fig. 5A uses the label “Logical Address” but that term is not used anywhere in the written description. And, as I noted, the term “physical address” is only used once. A POSITA would not make a mountain of this molehill.

e. 8[d]: “resetting the identifier value for the indicated first image frame of the independent sequence”

146. In my opinion, Kim teaches this limitation. Specifically, Kim’s Figure 5A shows that the IAD component of each logical address is reset to zero for each I-picture. (See EX1006, Fig. 5A (annotated below to show IAD component of each I-picture’s logical address in red and the whole logical address outlined in purple).) Because each I-picture is the first image frame of an independent sequence, Kim discloses that the IAD component of each logical address (“identifier value”) is reset for the indicated first image frame of the independent sequence.



(EX1006, Fig. 5A.)

147. To the extent Patent Owner disputes that Kim’s disclosure of resetting a component of an identifier is sufficient to meet the limitation, I would disagree for the reasons I discussed above in Section IX(A)(2)(d) in addressing the similar “resetting” limitation in 1[c].

148. Accordingly, in my opinion Kim teaches resetting the identifier value for the indicated first image frame of the independent sequence through its disclosure that the IAD portion of a picture’s logical address is reset for the first frame of an independent sequence.

149. In IPR2024-00691 (*i.e.*, the Amazon IPR case involving the ’321 patent), Patent Owner argued that the language in claim 8 requires the *decoder* to do the resetting whereas Kim discloses that the identifier for the first picture in the independent sequence is reset by the *encoder*. In my opinion, this argument mischaracterizes the ’321 patent and what element 8[d] requires. The ’321 patent states “when activated at a random point, the decoder preferably sets the identifier of the first image frame, preferably an I-frame, of the independently decodable sub-sequence at zero.” (EX1001,

11:37-40.)⁶ Importantly, however, the patent repeatedly and consistently teaches that the image number of the first image in each sub-sequence is zero. (EX1001, 8:19-22 (“Next, within each sub-sequence, the video frames of the sub-sequence are given image numbers, using for example consecutive numbering that starts with the number 0 given to the first video frame of the sub-sequence.”); 8:38-40 (“Further, since the I-frame 202 also forms a separate sub-sequence, i.e. it is the only video frame in the sub-sequence, its image number is 0.”); 8:57-59 (“A P-frame 204 begins the first sub-sequence of the base layer, and therefore the sub-sequence identifier of the P-frame 204 is 0.”); 9:13-15 (“Further, since the P-frame 208 is the first and only image frame in said sub-sequence, the P-frame 208 receives the image number 0.”); 9:21-22 (“Since the P-frame 210 is the first image frame in the sub-sequence, the image number of the P-frame 210 is 0.”); 9:35-37 (“Further, since the P-frame 214 is the first and only image frame in the sub-sequence, the image number of the P-frame 214 is 0.”).) Every I-frame and every P-frame that starts a sub-sequence in Figures 2 through 6 has an image number of zero. Accordingly, when the “decoder ... sets the identifier of the first image frame” to zero (EX1001, 11:37-40), a POSITA would know it does not change the

⁶ Although this disclosure uses the term “set” rather than “reset,” Nokia identified this disclosure during prosecution as the written description support for the resetting limitation in the decoder claims. (EX1002, 776.)

image number set by the encoder. Generally speaking, video decoders reconstruct the images based on the information provided by the encoder via the bitstream. Accordingly, a POSITA would not expect a decoder to receive an identifier value set by the encoder and then modify that value and nothing in the '321 patent suggests why a decoder would want or need to do this either.

150. In my opinion, a POSITA would have understood that, consistent with the '321 patent, Kim's decoder resets the identifier for the first image frame (an I-frame) of the independent sequence during decoding. Kim's decoder stores I-pictures in frame memory 101 for use in decoding subsequent P-pictures. (EX1006, 5:48-52, 5:55-59.) In my opinion, it would have been obvious to a POSITA for Kim's decoder to store the IAD along with the I-picture so the decoder could determine whether the I-picture needed to decode a given P-picture is already present in frame memory. Setting the IAD for the I-picture (*e.g.*, 0000 for the first I-picture in Fig. 5A) in frame memory 101 would set (or reset) the identifier in the same manner described by the '321 patent. (*See* EX1001, 11:37-40.) Nothing in the '321 patent, as it would have been understood by a POSITA, requires the decoder to change the identifier value assigned during encoding for the first frame in the sequence.

151. For the reasons discussed above, it is my opinion that Kim renders obvious claim 8.

8. Claim 10

152. In my opinion, Kim teaches claim 10.

153. Claim 10 is substantively the same as claim 8, except instead of being written in terms of “[a] method for decoding a compressed video sequence” it is written in terms of a “[a] video decoder comprising at least one processor and at least one memory including computer program code, the at least one memory and the computer program code configured to, with the at least one processor, cause the video decoder to decode a compressed video sequence.”

- a. **10[pre] “A video decoder comprising at least one processor and at least one memory including computer program code, the at least one memory and the computer program code configured to, with the at least one processor, cause the video decoder to decode a compressed video sequence, by causing the video decoder to at least”**

154. In my opinion, Kim teaches or suggests 10[pre]. As discussed in the Section IX(A)(7)(a) for 8[pre], Kim teaches “decoding a compressed video sequence.” Kim teaches the additional elements of 10[pre].

155. Kim discloses a Time Lapse recording/reproduction device that includes “a decoder 111 for performing a reverse-quantization and reverse-transformation of ... demultiplexed image data”; “a reproducing-DSP 121 for generating original image frames based on the P-picture data output from the decoder 111 and moving vector output from the de-MUX 131; and “a controller 150 for controlling the above-described components based on the commands from the user.” (EX1006, 3:18-27, Fig. 2.)

156. A POSITA would have understood that the disclosed controller 150 and corresponding decoder 111 and reproducing DSP 121 comprise at least one processor and at least one memory including computer program code configured to, with the at least one processor, cause the video decoders to decode a compressed video sequence. As I discussed in Section IX(A)(4)(a) for 4[pre], a DSP is a specialized type of computer processor that executes a stored program of computer instructions for performing signal processing operations. Thus, the disclosure of the DSP by itself is sufficient to teach “at least one processor and at least one memory including computer program code configured to, with the at least one processor, cause the video decoders to decode a compressed video sequence.”

157. Furthermore, Kim discloses encoding image frames in accordance with MPEG (EX1006, 4:15-19) and subsequently decoding those image frames (EX1006, 5:8-68). By the time of the alleged invention, software-based MPEG decoders were known and commercially available. (*See, e.g.*, EX1008 at 366 (identifying software-based MPEG decoders by vendor); *id.* at 367-381; *see also, e.g.*, EX1013 (description of C-Cube CL480PC MPEG-1 audio/video decoder).) By 1998, there were software-based reference encoders and decoders for MPEG-2. (EX1017.) Accordingly, it would have been obvious to a POSITA to implement an MPEG decoder with a processor, memory and computer code as claimed. Accordingly, in my opinion, it would have

been obvious to a POSITA to similarly implement Kim's decoder with a processor, memory and computer code as claimed.

- b. 10[a]: “decode from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence”**

158. In my opinion, this limitation is almost identical to limitation 8[a] and is obvious for the same reasons I discussed in Section IX(A)(7)(b).

- c. 10[b]: “start the decoding of the video sequence from said first image frame of the independent sequence, whereby the video sequence is decoded without prediction from any image frame decoded prior to said first image frame”**

159. In my opinion, this limitation is substantively identical to limitation 8[b] and is obvious for the same reasons I discussed in Section IX(A)(2)(c).

- d. 10[c]: “decode identifier values for image frames according to a numbering scheme”**

160. It is my opinion that this limitation is substantively identical to limitation 8[c] and is obvious for the same reasons I discussed in Section IX(A)(2)(d).

- e. 10[d]: “reset the identifier value for the indicated first image frame of the independent sequence”**

161. It is my opinion that this limitation is substantively identical to limitation 8[d] and is obvious for the same reasons I discussed in Section IX(A)(2)(e).

162. For the reasons discussed above, it is my opinion that claim 10 is unpatentable as obvious over Kim.

9. Claim 11

163. In my opinion, claim 11 is substantively the same as claim 8, except instead of being written in terms of “[a] method for decoding a compressed video sequence” it is written in terms of “[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence.”

- a. **11[pre]: “A computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence, the computer program product comprising”**

164. In my opinion, Kim teaches or suggests 11[pre]. As discussed in Section IX(A)(7)(a) for 8[pre], Kim teaches “decoding a compressed video sequence.” Kim also teaches the additional elements of 11[pre].

165. Kim discloses a Time Lapse recording/reproduction device that includes “a decoder 111 for performing a reverse-quantization and reverse-transformation of ... demultiplexed image data”; “a reproducing-DSP 121 for generating original image frames based on the P-picture data output from the decoder 111 and moving vector output from the de-MUX 131”; and “a controller 150 for controlling the above-described components based on the commands from the user.” (EX1006, 3:18-27.)

166. A POSITA, considering Kim’s Figure 2, would have understood the disclosed decoder 111, reproducing DSP 121, and controller 150 together or

individually to be “data processing devices,” given Kim’s disclosure that they all process data in the form of image data or commands.

167. A POSITA would have found it obvious to implement Kim’s decoder 111, reproducing DSP 121, and controller 150 as “[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence.” More specifically, a POSITA would have understood that Kim’s decoder 111 is “for performing a reverse-quantization and reverse-transformation of the demultiplexed image data” (EX1006, 3:18-20) and that reproducing DSP 121 is “for generating original image frames based on P-picture data output from the decoder 111 and the moving vector output from the de-MUX 131” (EX1006, 3:21-23), both under control of controller 150 which is “for controlling the above-described components based on the commands from the user.” (EX1006, 3:25-27).

168. As I discussed above in Section IX(A)(4)(a), a DSP is a specialized type of computer processor that executes a stored program of computer instructions for performing signal processing operations. Accordingly, a POSITA would have understood that Kim’s reproducing DSP 121 necessarily performs its portion of the decoding process by executing a “computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence.”

169. A POSITA also would have found it obvious to implement the remaining video decoding functions performed by Kim's decoder 111 and controller 150 in either of two ways: as dedicated custom integrated circuit hardware or as a stored computer program running on a general-purpose processor that was available at the time. Both of these implementation options would have been obvious to a POSITA and both would have been well within the capabilities of a POSITA. Therefore, in my opinion it would have been obvious to a POSITA to implement Kim's decoder 111, reproducing DSP 121, and controller 150 in one or more computer processors executing "[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence."

170. A POSITA would have been motivated to implement Kim in this way and would have had an expectation of success at least because, at the time, there were already well-known examples of implementing real-time MPEG video decoders in software computer program products running on general purpose computer processors. For example, a real-time software MPEG-1 decoder was available for free download from the University of California since at least 1993 (*see generally* EX1010) and engineers at Hewlett-Packard had implemented a real-time software MPEG-1 video decoder on HP workstations and personal computers at least by 1995 (EX1009, 60-68), with timing information for their software implementation of the video decoding functions performed by Kim's decoder 111 given in Table 1 (EX1009 at 63). Therefore,

a POSITA would have found it obvious to implement them in computer program software on a computer processor such that, together with reproducing DSP 121 (also a computer processor), they would meet the limitation “[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence, the computer program product.” Finally, a POSITA would have understood that these computer program products would have been stored on a non-transitory computer readable medium so that they could be stored and then executed by the one or more processors to decode a video sequence “based on the commands from the user.” (EX1006, 3:25-27).

171. Thus, based on the foregoing, it is my opinion that a POSITA would have understood Kim to disclose a “computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence.”

172. Furthermore, Kim discloses encoding image frames in accordance with MPEG (EX1006, 4:15-19) and subsequently decoding those image frames (EX1006, 5:8-68). By the time of the alleged invention, software-based MPEG decoders were known and commercially available. (*See, e.g.*, EX1011 at 366 (identifying software-based MPEG decoders by vendor); *id.* at 367-381; *see also, e.g.*, EX1007 (description of C-Cube CL480PC MPEG-1 audio/video decoder).) Accordingly, it would have been obvious to a POSITA to implement an MPEG decoder with a processor, memory and

computer code as claimed. By the same token, in my opinion, it would have been obvious to a POSITA to implement Kim's decoder as "[a] computer program product, stored on a non-transitory computer readable medium and executable in a data processing device, for decoding a compressed video sequence."

- b. **11[a]: "a computer program code for decoding from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence"**

173. In my opinion, this limitation is substantively identical to limitation 8[a], except that it is directed to "a computer program code for" decoding an indication rather than the act of decoding an indication. As I explained above in Section IX(A)(7)(b), Kim teaches "decoding from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence." And as I explained in Section IX(A)(9)(a), a POSITA would recognize that the decoding would be carried out by "a computer program product," which would include "computer program code." Kim therefore teaches or suggests this limitation in my opinion.

- c. **11[b]: “a computer program code for starting the decoding of the video sequence from said first image frame of the independent sequence, whereby the video sequence is decoded without prediction from any image frame decoded prior to said first image frame”**

174. In my opinion, this limitation is substantively identical to limitation 8[b], except that it is directed to “a computer program code for” starting the decoding rather than the act of starting the decoding. As I explained above in Section IX(A)(7)(c), Kim teaches “starting the decoding of the video sequence from said first image frame of the independent sequence, whereby the video sequence is decoded without prediction from any image frame decoded prior to said first image frame.” And as I explained above in Section IX(A)(9)(a), a POSITA would recognize that the decoding process would be carried out by “a computer program product,” which would include “computer program code.” Therefore, it is my opinion that Kim teaches or suggests this limitation.

- d. **11[c]: “a computer program code for decoding identifier values for image frames according to a numbering scheme”**

175. It is my opinion that this limitation is substantively identical to limitation 8[c], except that it is directed to “a computer code for” decoding identifier values rather than the act of decoding identifier values. As I explained above in Section IX(A)(7)(d), Kim teaches “decoding identifier values for image frames according to a numbering scheme.” And as I explained above in Section IX(A)(9)(a), a POSITA would recognize that the decoding process would be carried out by “a computer program product,” which

would include “computer program code.” Kim therefore teaches or suggests this limitation in my opinion.

e. 11[d]: “a computer program code for resetting the identifier value for the indicated first image frame of the independent sequence”

176. In my opinion, this limitation is substantively identical to limitation 8[d], except that it is directed to “a computer program code for” resetting the identifier value rather than the act of resetting the identifier value. As explained in Section IX(A)(7)(e), it is my opinion that Kim therefore teaches “resetting the identifier value for the indicated first image frame of the independent sequence.” And as I explained above in Section IX(A)(9)(a)IX.A.9.a, a POSITA would recognize that the decoding process would be carried out by “a computer program code product,” which would include “computer program code.” Kim therefore teaches or suggests this limitation in my view.

177. For the reasons discussed above, it is my opinion that Kim teaches or suggests claim 11.

B. Ground 2: Kim in view of Adolph Render Obvious claims 2, 5, and 9

1. Adolph

178. Adolph discloses a video coding technique for reducing resynchronization delays by inserting important picture-level information into the slice header of a video sequence. (EX1008, Abstract.) Adolph explains that channel interference can cause data loss and when important data is lost, “undesirably long failure times [can] arise in

the decoder which is used, as a result of consequential errors occurring.” (EX1008, 1:11-21.)

179. Adolph proposes modifications of an MPEG bit stream “to achieve a shorter resynchronization time and a reduction in consequential errors.” (EX1008, 1:32-35.) “[A]dditional information is inserted, according to the invention, at an advantageous point into the slice header, which information contains the most important information about the respective frame and thus allows decoding even in the event of loss of the picture start code or extension start code” (EX1008, 2:3-11.) The additional information can “relate[] ... to the frame type (I-, P-, or B-frame) and/or to important parameters for decoding of movement vectors.” (EX1008, 2:41-45.) Adolph’s technique “complies with the MPEG standard, in particular the MPEG2 standard.” (EX1008, 2:31-33.)

180. Adolph’s Figure 3 (below) shows additional information added to the slice header including the “temporal[] reference, in which an integer number indicates the chronological arrangement of the encoded frame” and the “picture coding type, which indicates whether the frame is I-, P-, or B-encoded.” (EX1008, 4:1-15.)

<u>EXTRA INFORMATION SLICE</u>	
<u>PICTURE HEADER:</u>	BITS
TEMPORAL_REFERENCE	8
PICTURE_CODING_TYPE	2
<u>PICTURE CODING EXTENSION:</u>	
FORWARD_HORIZONTAL_f_CODE	4
FORWARD_VERTICAL_f_CODE	4
BACKWARD_HORIZONTAL_f_CODE	4
BACKWARD_VERTICAL_f_CODE	4
INTRA_DC_PREDICTION	2
CONCEALMENT_MOTION_VECTORS	1
q_SCALE_TYPE	1
INTRA_VLC_FORMAT	1
ALTERNATE_SCAN	1

INFORMATION CONTENTS OF THE SLICE HEADER

FIG. 3

(EX1008, Fig. 3 (annotated); 4:1-15.)

181. Adolph teaches including this additional information in the “extra_information_slice” field that the MPEG standard already provides for in the slice header. (EX1008, 3:60-63; Fig. 2.) Consistent with MPEG-2 (*see* EX1016 at 30), Adolph’s Fig. 2 shows that the size of the extra_information_slice is flexible:

while (nextbits () == '1') {		
extra_bit_slice /* with the value "1" */	1	uimbsf
extra_information_slice	8	
}		

(EX1008, Fig. 2 (excerpted).) To increase coding efficiency, the information may be included only in certain slices (e.g., every other macroblock line or every third macroblock line). (EX1008, 2:37-41.)

182. A decoder can use the additional information provided in the slice header to resynchronize even if the picture header is lost due to transmission errors. (EX1008, 5:19-27.) As a result, “[l]ong resynchronization times are avoided ... and consequential errors in the decoding resulting from fault predictions are reduced.” (EX1008, 5:26-29.)

2. Motivation to Combine Kim and Adolph

183. It would have been obvious to a POSITA in my opinion to combine the teachings of Kim and Adolph so as to include Kim’s “most important” picture-level decoding information in the slice header.

184. Kim teaches or suggests the use of MPEG in encoding video data. Moreover, by the time of the alleged invention, MPEG was well-established as a leading video compression standard. (*See, e.g.*, EX1022 (discussing success of MPEG-1 and MPEG-2 standards.)

185. Adolph describes a straightforward error resilience technique to reduce resynchronization time by including certain critical picture-related data in slice headers in an MPEG bitstream. Kim discloses sequences of video frames consisting of an I frame followed by one or more P frames that are differentially encoded relative to the I frame. Kim further discloses picture-level logical address information that indicate the

corresponding I frame for each P frame and that this information is used during decoding. In particular, a given P frame cannot be decoded without first decoding its corresponding I frame and that I frame is indicated in the address information for the P frame. (See discussion above in Sections IX(A)(1) and IX(A)(7)(d).) Because in Kim's approach *every* P-picture in the independent sequence relies on the same I-picture, data loss related to the I-frame (e.g., it's CAD), which could occur, e.g., due to a disk read error or a failure of the optical disk media, would have drastic impacts on the decoding of the rest of the frames in the sequence. Adolph's error resilience technique would provide information that could allow the decoder to decode the sequence properly even if some picture-level information was lost or corrupted.

186. In my opinion, it would have been obvious to a POSITA to apply Adolph's teachings by inserting Kim's "most important" picture-level information (*i.e.*, the picture's IAD and CAD) in the slice headers as taught by Adolph. To reduce the level of redundancy and improve coding efficiency, the information could be included only in a subset of slice headers as Adolph teaches.

187. The '321 patent does not define the term "flag" or provide any examples of a flag in the slice header. In my opinion, a POSITA at the time of the invention would have considered a zero value IAD of 0000 (as Kim discloses) to be a separate flag as claimed. There were known examples of multi-bit flags in video standards. One example is the 3-bit Reference Picture Selection Mode Flags (RPSMF) in H.263. But

even if Nokia argues that the claimed flag needs to be a *single-bit* flag, this still would have been obvious in my opinion. Adolph specifically teaches inserting the picture type in the slice header. (EX1008, 2:41-45, 4:5-7; Fig. 3.) Adolph also teaches minimizing the number of bits used to represent the information added to the slice header. (EX1008, 2:8-11, 2:45-51, 4:7-15.) Given that Kim's coding method uses only two picture types (I- and P- pictures), it would have been obvious to a POSITA that Kim's picture type could be represented with a one-bit flag (e.g., an 'I-picture' flag). A value of '1' could indicate the current picture is an I-picture and a value of '0' could indicate that the current picture is a P-picture. In my opinion, a POSITA would have recognized that the use of a single-bit 'I-picture' flag would efficiently allow for the picture type to be coded with a single bit in the slice header in order to minimize the overhead associated with the information added to the slice headers. A POSITA would have understood that such a single-bit "I-picture" flag would provide an equivalent, but more efficient, way of indicating an I-frame than using multiple zero bits to indicate each I-frame as Kim describes and shows in Figs. 5A, 5B, and 6.

188. In my opinion, a POSITA would have had a reasonable expectation of success in combining the teachings of Kim and Adolph in this way. Adolph's technique is MPEG-compliant and utilizes an available field in the slice header to carry the additional information. If a generic MPEG decoder is not designed to use the additional information in the slice header, Adolph teaches the decoder will simply ignore the

additional information. (EX1008 at 5:29-32.) The proposed combination would provide the same information Kim describes to facilitate Kim's decoding method.

189. Kim describes time lapse recording systems which store compressed digital video on optical disks. (EX1006, 1:8-11.) Adolph suggests application of its error resilience techniques to systems using "digital storage media such as DVD." (EX1008, 5:36-38.)

190. To the extent Patent Owner argues that a POSITA would not be motivated to apply Adolph's technique to Kim because it would reduce coding efficiency by adding redundant information at the slice layer, I disagree. The premise underlying Adolph's scheme is that the error resilient benefits justify the added overhead of providing some redundant information at the slice layer. A POSITA would have readily understood the tradeoffs inherent in Adolph's technique and that the error resilience benefits would justify the reduction in coding efficiency at least in certain applications. Moreover, since Kim discusses using optical discs which have a maximum storage capacity which may not be fully utilized to store the compressed video data, a POSITA would have understood that Kim's time lapse video system would be well-suited to Adolph's teachings for improving error resilience.

3. Claim 2: "A method according to claim 1, further comprising: encoding the indication into the video sequence as a separate flag included in the header of a slice."

191. In my opinion, Kim teaches claim 2.

192. As I just discussed, it would have been obvious in my opinion from Kim’s and Adolph’s teachings to include Kim’s I-picture IAD indicator (0000) in the slice header as a separate flag. As I also discussed, even if Nokia argues that the claimed flag needs to be a single-bit flag, it also would have been obvious to use a one-bit flag (*e.g.*, an “I-picture” flag) in the slice header to distinguish between Kim’s two picture types. A one-bit flag would have been consistent with Adolph’s teachings of (1) indicating the frame type in the slice header (EX1008, 2:41-45, 4:5-7; Fig. 3), and (2) minimizing the number of bits used for the information added to the slice header (EX1008, 2:8-11, 2:45-51, 4:7-15). Moreover, this would have provided a more efficient way of signaling the picture type than Kim’s approach of using a IAD value having *multiple* zero bits. (EX1006, Figs. 5A, 5B, 6.)

193. Accordingly, Kim and Adolph render obvious claim 2.

4. **Claim 5: “A video encoder according to claim 4, wherein the video encoder is configured to encode the indication into the video sequence as a separate flag included in the header of a slice.”**

194. Claim 5 adds an almost identical limitation to claim 2 and is obvious for the same reasons I discussed in Section VII(B)(3).

5. **Claim 9: “A method according to claim 8, wherein the indication is a separate flag included in the header of a slice.”**

195. Claim 9 adds an almost identical limitation to claim 2 and is obvious for the same reasons I discussed in Section VII(B)(3).

X. CONCLUSION

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

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

Date: June 23, 2025

By:


Dr. Joseph P. Havlicek