IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

AMAZON.COM, INC., AMAZON.COM SERVICES LLC,

Petitioner,

v.

NOKIA TECHNOLOGIES OY,

Patent Owner.

_____

Case No. IPR2024-01503

U.S. Patent 9,571,833

_____

Declaration of Charles D. Creusere

TABLE OF CONTENTS

Page

I. OVERVIEW OF THE TECHNOLOGY 5
- A. Video Compression Basics 5
- B. Motion Vectors and Motion Prediction 6
- C. H.264 to H.265 12

II. THE '833 PATENT 15
- A. Overview 15
- B. Prosecution History 17
- C. Priority Date 20
- D. Challenged Claims 20

III. LEVEL OF ORDINARY SKILL IN THE ART 21

IV. CLAIM CONSTRUCTION 22
- A. "spatial motion vector prediction candidate" 22
- B. "temporal motion vector candidate" 23
- C. "the block" 24
- D. "a subset of … candidates" 25

V. INVALIDITY GROUNDS 25
- A. Ground 1 27
  1. Rusert (Ex-1004) 27
  2. Zheng (Ex-1005) 30
  3. Independent Claim 9 34
  4. Dependent Claim 10 101
  5. Dependent Claim 11 103
  6. Dependent Claim 12 116
  7. Dependent Claim 13 119
  8. Dependent Claim 14 136
  9. Independent Claim 16 138

10. Independent Claim 18 144
11. Dependent Claim 26 150
12. Dependent Claim 27 150
13. Dependent Claim 28 151
14. Dependent Claim 29 151
15. Dependent Claim 30 153
B. Ground 2 154
1. Nakamura (Ex-1007, Ex-1008, Ex-1009) and WD4 (Ex-1010) 154
2. Motivation to Combine and Reasonable Expectation of Success 157
3. Independent Claim 9 165
4. Dependent Claim 10 191
5. Dependent Claim 11 193
6. Dependent Claim 12 199
7. Dependent Claim 13 204
8. Dependent Claim 14 215
9. Independent Claim 16 216
10. Independent Claim 18 220
11. Dependent Claim 26 226
12. Dependent Claim 27 226
13. Dependent Claim 28 227
14. Dependent Claim 29 228
15. Dependent Claim 30 230
VI. BACKGROUND AND QUALIFICATIONS 230
A. Compensation 235
B. Materials and Other Information Considered 235
VII. UNDERSTANDING OF THE LAW 235
VIII. RESERVATION OF RIGHTS 235
APPENDIX 1: CURRICULUM VITAE OF CHARLES D. CREUSERE 237

APPENDIX 2: MATERIALS CONSIDERED IN THE PREPARATION OF THIS DECLARATION 252
APPENDIX 3: CHALLENGED CLAIMS 257
APPENDIX 4: UNDERSTANDING OF THE LAW 270

I, Charles D. Creusere, declare as follows:

1. My name is Charles D. Creusere. I am a Full Professor in the Klipsch School of Electrical & Computer Engineering at New Mexico State University. I have prepared this report as an expert witness retained by Amazon.com, Inc. and Amazon.com Services LLC. In this report, I give my opinions as to whether certain claims of U.S. Patent No. 9,571,833 ("the '833 patent") are invalid. I provide technical bases for these opinions as appropriate.

2. This report contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein.

I. OVERVIEW OF THE TECHNOLOGY

A. Video Compression Basics

3. Video comprises a sequence of pictures, called frames. If you look closely at a frame, for example by putting a magnifying glass up to your monitor, you will see that a frame is comprised of pixels. Groups of pixels in a frame are referred to as blocks, for example a square of 8 pixels by 8 pixels.

4. Video encoding, also referred to as video compression, exploits redundancies in video data to reduce the number of bits required to store or

transmit the video. Since the 1990s, major video coding standards, including MPEG, H.264 ("AVC"), and H.265 ("HEVC"), have applied the same block-based model: video frames are divided into blocks of pixels, a motion estimation and compensation front end exploits temporal and spatial patterns where a similar-looking block appears in the same frame or a nearby frame, and then a coded bitstream is produced in subsequent stages that include transformation, quantization, and entropy encoding. Ex-1023, 000004. At the decoder, the inverse process is used to reconstruct the compressed video data back into the original frames.[1] *Id.* A high-level conceptual diagram of encoder stages is illustrated below. *See e.g.*, Ex-1023, Fig. 2:

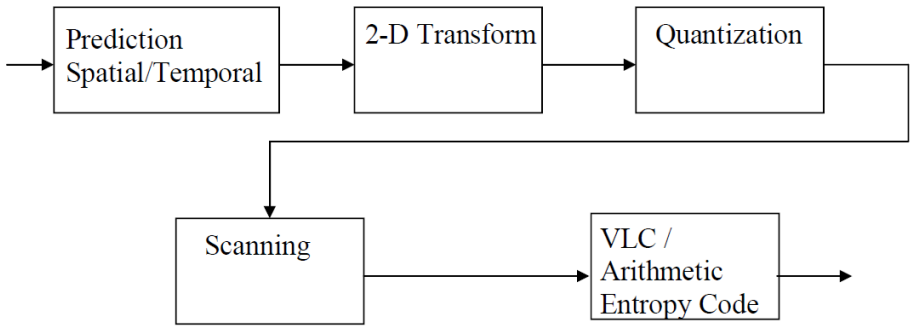


Fig. 2: Higher-level encoder block diagram

B. Motion Vectors and Motion Prediction

5. Blocks of pixels could be encoded using inter or intra-frame modes. *Intra*-frame encoding exploits redundant patterns within the same frame, with a

[1] Many of the video coding concepts, including HEVC concepts, described in this declaration still apply today.

block possibly being encoded with reference to another block in the that frame. *Inter*-frame coding allows a block to be encoded with reference to similar blocks in other frames.

6. For example, inter-picture prediction is useful when an object moved across the screen in successive frames. In the example below, the same airplane on the bottom left of one frame appears on the top-right of the next frame. Ex-1021, 000002 (emphasis added):

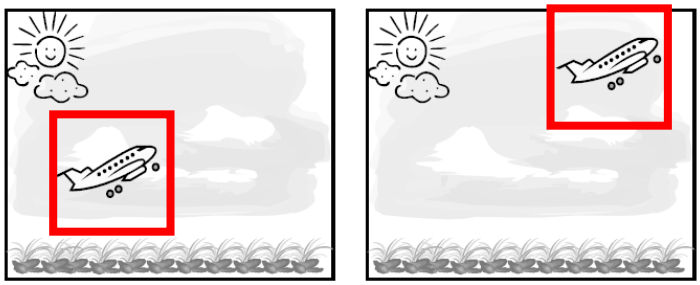
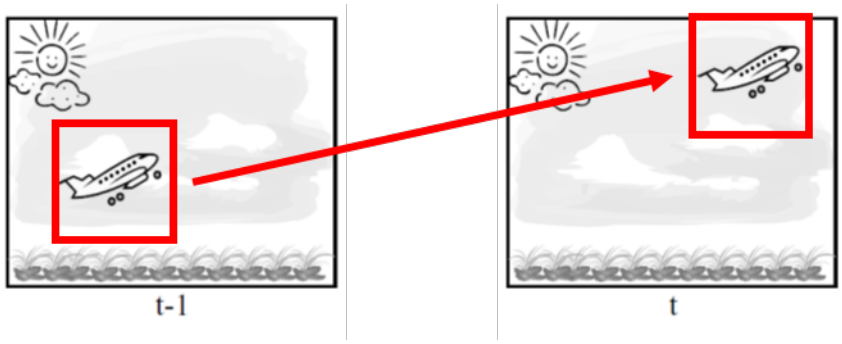


Fig. 3. Successive video frames

7. Since the block containing the airplane shifted to the right (in the X direction) and upwards (in the Y direction), a motion vector was used to describe this X-Y displacement. Ex-1021, 000002-4; Ex-1022, 000001-3. In other words, motion vectors describe the motion of blocks between frames. Rather than transmit the pixels for this block twice (once for each frame), the video coder instead transmits the block once—for the first frame—and then signals a motion vector, an index to a reference block in the first frame, and the residual difference between the motion compensated first frame block and the current frame block.

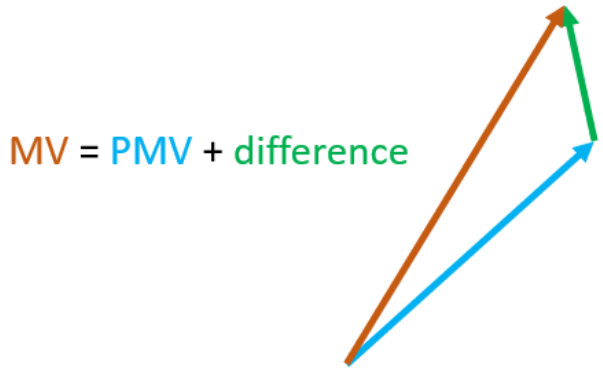
The decoder would then use this information to reconstruct the block of pixels containing the airplane in the second frame using the referenced block from the first frame. *See e.g.*, Ex-1021, 000004, Fig. 3 (annotations in red added):



8. By sending a motion vector, a block index, and the residual difference, rather than an entire block of pixels, video coders reduce the amount of data in the encoded video stream. Nonetheless, since each frame contains many thousands of blocks and with frames streaming at 30 or 60 frames a second, the volume of data required for transmitting motion vectors could be significant. *See* Ex-1022, 000001.

9. To reduce the amount of data used to signal motion vectors, the ITU utilized motion vector prediction in its H.264 standard as well as in the successor standard that became H.265. Ex-1022, 000001-2. For example, early iterations of H.264 used the median values of motion vectors from three spatially-neighboring blocks as a Predicted Motion Vector ("PMV") or motion vector predictor ("MVP"). *Id.* Since the encoder and decoder could independently calculate this PMV, there was no need to transmit the entire vector; instead, a smaller difference

vector was inserted into the video stream to indicate the difference between the predicted and actual motion vector for a block, as illustrated below. *Id.*



10. By 2010, neighboring blocks were commonly used to find PMV candidates. This was known for H.264. Ex-1004, ¶¶2-3. Since early drafts of H.265, by at least Working Drafts 3 and 4, multiple spatial motion vector prediction candidates were obtained from neighboring blocks, labelled A_0, A_1, B_0, B_1, B_2 below, to create a list of candidates. The encoder evaluated which motion vector candidate offered the best prediction for the current block—the one whose predicted pixel values were closest to the actual pixel values. Ex-1010, §8.4.2.1.8 (annotated with the current block):

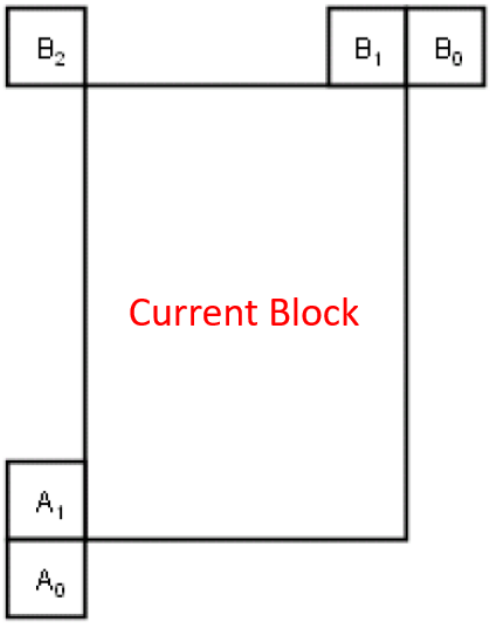


Figure 8-5 – Spatial motion vector neighbours

Since the encoder and decoder independently generated the same candidate list (using the same information from neighboring blocks), the encoder simply signaled to the decoder which candidate was chosen from the list, for example sending an index value of 2 to signal the second candidate from the list. This reduced the amount of data because the encoder did not need to transmit the candidate list or a motion vector; instead, the encoder transmitted a single, small number to signal which candidate was chosen, and the decoder referred to its independently-constructed candidate list, which was identical to that of the encoder, to look up the candidate using the transmitted index. *See e.g.*, Ex-1004, ¶37.

11. Motion vector candidates from the same frame, including those from neighboring blocks in the frame, were called spatial motion vector prediction

candidates. Ex-1022, 000001-3. Neighboring blocks were often good sources of motion vector candidates because neighboring blocks tend to have the same motion. For example, when a scene pans to the right, objects in the background will tend to have similar motion towards the left across the screen. Ex-1022, 000001-2. Predicted motion vectors take advantage of such patterns to decrease the amount of data used to encode motion vectors, as explained above.

12. Redundant motion vector prediction candidates in the list, however, can increase the size of the index needed to signal which candidate was selected as the predictor for a given block. *E.g.*, Ex-1004, ¶7, ¶107 (explaining that more bits are "needed to specify the candidate vector" as the number of vectors grows). For example, if the neighboring blocks all have the same motion vector, it would be redundant to include all of them as motion vector predictor candidates. Therefore, H.264 and H.265 both analyzed spatial motion vector prediction candidates to remove redundant ones. In fact, it was common in the art to remove redundant motion vector candidates. For example, this was taught for H.264 and in working drafts of H.265. Ex-1004, ¶21, ¶62; Ex-1010, §8.4.2.1.1, §8.4.2.1.7.

13. For H.264, prior art that applied H.264 included methods for selecting PMV candidates from the set of all previously-coded blocks. Since "[l]imiting and/or reducing the number of candidates … can be helpful to reduce the overhead of signaling," it was known in the art to "avoid duplicate occurrences of the same

motion vector" by "comparing the candidates already in the list with the new vector that could be added[,]" which comprised a "subset" of candidates from blocks within an allowed distance from the current block, rather than the full set of previously-coded blocks. Ex-1004, Abstract, ¶¶70-71.

14. For H.265, presentations were made to the standards body regarding the exact manner in which redundant candidates might be removed. "When motion vectors have the same value, the motion vectors are removed from the list [of motion vector candidates.]" Ex-1010, §8.4.2.1.7, §8.4.2.1.1. As I will explain in more detail below, at the 6th HEVC meeting held on July 14-22, 2011, Nakamura presented a proposal, JCTVC-F419, to reduce the number of comparisons needed for removing redundant motion vector candidates. *Infra* §V.B; Ex-1007, Fig. 1. Nakamura proposed creating a subset of two spatial motion vector prediction candidates, so that redundant candidates could be identified and removed with fewer comparisons and without having to compare the full set of neighboring candidates. Ex-1007, 000003, Table 1, Fig. 1.

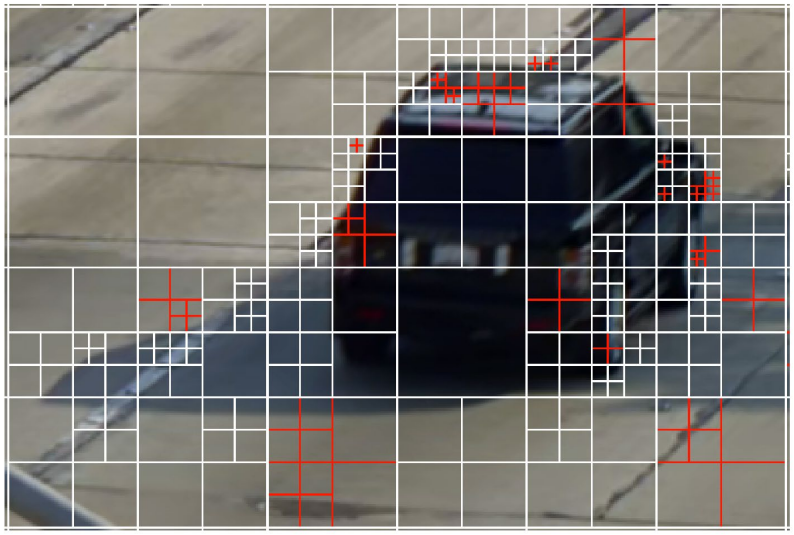
C. H.264 to H.265

15. The H.264 standard (also standardized by the ISO as MPEG 4 Advanced Video Coding ("AVC")) was published in 2003. H.264 was widely used for compression of video at HD (high definition) resolutions and below, and from the beginning it was popular in consumer electronics and Internet streaming.

In 2010, the standardization process began for the successor video standard, which was called H.265 by the ITU and also referred to as the ISO High Efficiency Video Coding ("HEVC") standard.

16. H.265 used H.264 as a starting point and incorporated many of the same concepts. Both standards followed the same overall architecture for block-based video coding, as explained above. Both also used predicted motion vectors for motion compensation. Ex-1004, ¶3; Ex-1011, ¶3; Ex-1005, ¶¶36, 39, Ex-1056, 000016-18 (¶¶20-25); Ex-1006, ¶4.

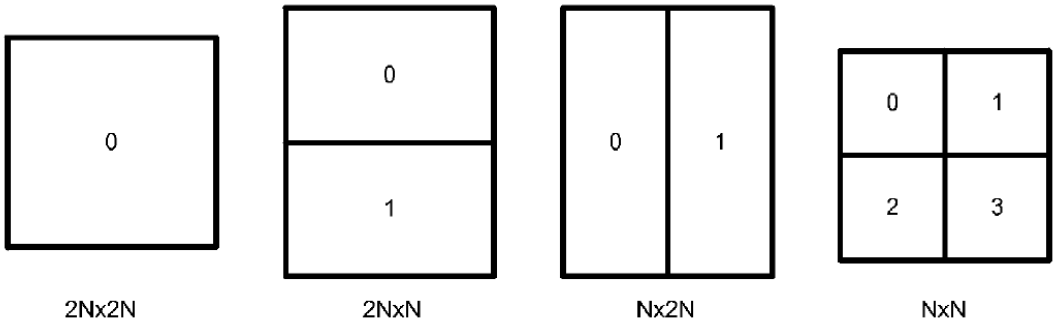
17. HEVC introduced new terminology for a type of block called a "coding unit" or "CU," which was analogous to the macroblocks described in H.264. *See* Ex-1011, ¶¶32-33, ¶64 ("block" generally "may refer to one or more of a macroblock, LCU, CU, sub-CU, TU, or PU."); Ex-1005, ¶28; Ex-1056, 000016 (¶¶17, 19); Ex-1006, ¶4. CUs could be split into smaller CUs to fit visual patterns in the frame. "The basic partition geometry of all these elements is encoded by a scheme similar to the well-known quad-tree segmentation structure." *See* Ex-1019, 000005; Ex-1020 at 000002 (Fig. 1):



18. Motion vectors operated on the basis of a block called a "prediction unit" or "PU." Ex-1011, ¶35; Ex-1005, ¶36; Ex-1056, 000016-17 (¶20); Ex-1006, ¶25. In the simplest scenario, one CU would be covered by a PU of the same size, and a single motion vector would be generated for the entire CU/PU. *See* Ex-1011, ¶35. This approach was used when the entire CU moved in the same direction, such as large blocks of road in the above image. Conversely, when different portions of a CU moved in different directions, the CU could be divided into multiple Pus, with different motion vectors assigned to each PU. *See id*.

19. HEVC referred to an undivided CU (consisting of one PU the same size) as being 2Nx2N. When a CU was split horizontally into two symmetric rectangular PUs, it was referred to as a 2NxN partition size. A vertically divided CU was referred to as an Nx2N partition size. Example partition modes are shown

below. Ex-1005, ¶47, Fig. 2; Ex-1055, 000003 (Fig. 2); Ex-1006, ¶4; Ex-1056, 000003 (Fig. 2), 000018 (¶¶26-27); Ex-1018, ¶3, Fig. 2.



II. THE ’833 PATENT

A. Overview

20. The ’833 patent is directed to video encoding and decoding in the context of H.263, H.264, and Working Draft 4 of H.265/HEVC, which the specification admits were pre-existing video codecs that pre-dated the ’833 patent. Ex-1001, 1:29-41, 2:10-16.

21. The patent admits that prior video codecs “create[d] motion vector predictions” (“MVP”) by “generat[ing] a list or a set of candidate predictions from blocks in the current frame and/or co-located or other blocks in temporal reference pictures and signalling[sic] the chosen candidate as the motion vector prediction.” Ex-1001, 2:61-3:8, 3:49-53. “A spatial motion vector prediction is a prediction obtained only on the basis of information of one or more blocks of the same frame than the current frame whereas temporal motion vector prediction is a prediction

obtained on the basis of information of one or more blocks of a frame different from the current frame." *Id.*

22. Similarly, the '833 patent refers to the merge mode, which was a mode of operation in H.265 where the list of candidates was called a "merge list" instead of an MVP list. The '833 patent describes "a merge process for motion information coding" that "creates a list of motion prediction candidates" in which a "maximum number of spatial motion prediction candidates to be included in the merge list may be defined," and provides as an example where the maximum number of spatial candidates in the merge list is four. Ex-1001, 14:24-51.

23. Regardless of what the list is called, "[a]fter the list is generated, some of the motion vector prediction candidates may have the same motion information. In this case, the identical motion vector prediction candidates may be removed to reduce redundancy." Ex-1001, 3:55-59. To find redundant candidates, the candidates in the list must be compared to each other.

24. The '833 patent seeks to "improv[e] the prediction accuracy and hence possibly reducing information to be transmitted in video coding systems." Ex-1001, 8:13-20. To this end, the '833 patent purports to introduce "a method for generating a motion vector prediction list" in "a way to reduce the complexity of the implementation." Ex-1001, 4:7-12. According to the '833 patent, "[t]his can be achieved by performing a limited number of motion information comparisons

between candidate pairs to remove redundant candidates rather than comparing every available candidate pair." Ex-1001, 4:12-15. In particular, the '833 patent claims recite limitations for "determining a set" of spatial MVP candidates, "determining a subset" of spatial MVP candidates, and then "comparing motion information" of a selected candidate with the subset, "without making a comparison of each possible candidate pair from the [full] set of spatial motion vector prediction candidates[.]" Ex-1001, claim 1.

25. However, as explained above, the prior art already included techniques for removing redundant candidates without comparing each pair from the full set of candidates. *Supra* §I.

B. Prosecution History

26. I understand the '833 patent is the start of a chain of a patents, including U.S. Patent No. 10,536,714, which is the subject of other *Inter Partes* Review proceedings (IPR2024-00604, IPR 2024-00605).

27. At the same time the '833 patent was being prosecuted in the United States, a European counterpart, EP2774375 (the "EU Counterpart"), was being prosecuted in the EP. Ex-1017, 000246-256 (extended European search report dated March 14, 2016); Ex-1002, 000960-981 (Response after Final Action dated July 1, 2016).

28. In the EP prosecution, an extended European search report issued on March 21, 2016, in which the European Patent Office issued a rejection in view of Nakamura. Ex-1017, 000246-256 (extended European search report). The Patent Owner did not dispute that Nakamura was prior art nor that Nakamura taught the claims. Instead, the Patent Owner amended the claims with lengthy limitations that led to independent claim 1 being over *two pages long* in the EP counterpart. Ex-1017, 000335-351.

29. On April 25, 2016, the Patent Owner submitted an Information Disclosure Statement to the US Patent Office during the prosecution of the ’833 patent citing the extended European search report and Nakamura. Ex-1002, 000896. The Examiner did not consider this IDS before the Notice of Allowance. Ex-1002, 001155-56 (mailed 12/20/2016); Ex-1002, 001163. The ’833 patent was allowed without any substantive discussion of Nakamura or amendments like those made in the EU counterpart. Ex-1002, 001033.

30. During prosecution, the Examiner issued rejections on the basis of three academic papers: “Improved Video Compression Efficiency Through Flexible Unit Representation and Corresponding Extension of Coding Tools” by Han et al., “A Multi-Pass True Motion Estimation Scheme with Motion Vector Propagation for Frame Rate Up-Conversion Applications” by Tai et al., and “A Multistage Motion Vector Processing Method for Motion-Compensated Frame

Interpolation" by Huang et al. Ex-1002, 000184-203 (non-final rejection based on Han and Tai), 000795-826 (final rejection based on Han, Tai, and Huang).

31. After a final rejection, on September 1, 2016, the Applicant filed a Request for Continued Examination and included claim amendments that recited: "comparing motion information of the first spatial motion vector prediction candidate with motion information of spatial motion vector prediction candidates in the determined subset of spatial motion vector prediction candidates **without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates** …" Ex-1002, 000998, 001002, 001005, 001006, 001007, 001008, 001009, 001010.

32. In the Notice of Allowance, the Examiner cited, as the reason for allowance, the limitation for comparing motion information of candidates "without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates[.]" Ex-1002, 001045.

33. However, the European Patent Office found a similar limitation to be taught by Nakamura (Ex-1017, 000253), *which the Patent Owner did not dispute.* Nakamura is presented here as Ground 2.

34. On October 23, 2018, the US Patent Office issued a Certificate of Correction, that corrected in the claims, column 33, line 56, specifically in claim 9, from: "comparing motion information of the first spatial motion vector prediction

candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates" to "comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the ***subset*** of spatial motion vector prediction candidates." Ex-1001, 33:53-56 (emphasis added) (incorporating Certificate of Correction); Ex-1002, 001180.

C. Priority Date

35. The '833 patent claims priority to a provisional application filed on November 4, 2011. I have not conducted any analysis as to whether the '833 patent is entitled to its claim of priority. For the purposes of this declaration, I have applied a November 4, 2011, priority date for my opinions without regard to whether the '833 patent is entitled to such a priority date.

D. Challenged Claims

36. I understand that Petitioner is challenging the validity of claims 9-14, 16, 18, 26-30 of the '833 patent in the Petition for *Inter Partes* Review to which this declaration will be attached. Those claims are reproduced in Appendix 3. While this Petition and declaration are directed to the challenged claims, I have considered all claims of the '833 patent, as well as portions of the '833 patent prosecution history in forming my opinions.

III. LEVEL OF ORDINARY SKILL IN THE ART

37. I have analyzed the ’833 patent and determined that the field of the patent is video encoding and decoding. For example, the ’833 patent characterizes its alleged invention as “a method for encoding, a method for decoding,” and “an encoder and a decoder.” Ex-1001, 1:6-8. More specifically, the ’833 patent states “[t]he present invention introduces a method for generating a motion vector prediction list for an image block.” Ex-1001, 4:7-8.

38. In determining the characteristics of a hypothetical person of ordinary skill in the art (“POSITA”) of the ’833 patent at the time of the claimed invention, I considered several things, including various prior art techniques relating to video encoding and decoding, the type of problems that such techniques gave rise to, and the rapidity with which innovations were made. I also considered the sophistication of the technologies involved, and the educational background and experience of those actively working in the field at the time. I also considered the level of education that would be necessary to understand the ’833 patent. Finally, I placed myself back in the relevant period of time and considered the engineers and programmers that I have worked with and managed in the field of video encoding and video decoding.

39. I came to the conclusion that a person of ordinary skill in the field of art of the ’833 patent would have been a person with (1) a 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. I at least qualify under the definition of a POSITA because I had a BS degree in electrical and computer engineering by 1985, with many years of experience in video encoding and decoding (more than three) by 2011, as explained in my qualifications section. *Infra* §VI.

IV. CLAIM CONSTRUCTION

40. For purposes of this inter partes review, I have considered the claim language, specification, and portions of the prosecution history to determine the meaning of the claim language as it would have been understood by a person of ordinary skill in the art at the time of the invention. The "plain and ordinary meaning" or Phillips standard has traditionally been applied in district court litigation, where a claim term is given its plain and ordinary meaning in view of the specification from the view point of a person of ordinary skill in the art.

41. I have applied the *Phillips* standard in my analysis. Unless otherwise stated, I have applied the plain and ordinary meaning to claim terms.

A. "spatial motion vector prediction candidate"

42. Based on my review of the claims and specification of the '833 patent, it is my opinion that a POSITA would have understood a "spatial motion vector

prediction candidate" to mean a "candidate motion vector obtained from one or more previously-encoded blocks in the current frame."

43. The specification of the '833 patent states that "a spatial motion vector *prediction* is a prediction obtained only on the basis of information of one or more blocks of the same frame than the current frame." Ex-1001, 2:65-3:3. Furthermore, the specification "defines candidate motion vectors for the current frame by using… one or more neighbour blocks and/or other blocks of the current block *in the same frame*…" Ex-1001, 12:38-43. Therefore, a spatial motion vector prediction *candidate* would be a candidate motion vector obtained only on the basis of information of one or more blocks of the current frame. *See* Ex-1001, 2:65-3:3, 12:38-43. The specification further states that spatial motion vector prediction candidates are obtained from "one or more already encoded block." Ex-1001, 12:45-46. That is, because spatial motion vector prediction candidates are "define[d]… by using one or more of the motion vectors of one or more neighbour blocks and/or other blocks of the current block in the same frame[,]" each spatial motion vector prediction candidate "represents the motion vector of one or more already encoded block." Ex-1001, 12:38-46.

B. "temporal motion vector candidate"

44. Based on my review of the claims and specification of the '833 patent, it is my opinion that a POSITA would have understood a "temporal motion vector

prediction candidate" to mean a "candidate motion vector obtained from a previously-encoded frame."

45. The specification of the '833 patent states that a "temporal motion vector *prediction* is a prediction obtained on the basis of information of one or more blocks of a frame different from the current frame." Ex-1001, 2:65-3:3. Furthermore, the specification describes that "for temporal prediction… motion vectors of a co-located block or other blocks in a previously encoded frame can be selected as candidate predictors for the current block." Ex-1001, 12:50-54; *see also id.* 12:54-57.

C. "the block"

46. Limitation [9d] recites "the block," which could be interpreted to refer to either (a) the "block of pixels" introduced in [9a], or (b) the block from which the first spatial motion vector candidate is obtained. It is my understanding that use of the word "the" refers back to something introduced earlier. Here, a POSITA would have understood that "the block" refers back to the "block of pixels" introduced in [9a] because there is no other block mentioned in the claim. This is consistent with the file history. During prosecution of the '833 patent, the Examiner applied the first interpretation, with "the" block as the "current block" (Ex-1002, 000198), which the Applicant did not dispute (Ex-1002, 000785-788).

For purposes of this IPR, Petitioner applies the Examiner's interpretation, where "the block" finds antecedent basis in the "block of pixels" in [9a].

D. "a subset of … candidates"

47. Limitations [9d], [16d], and [18d] recite "a subset of … candidates," which means a subset of one or more candidates. The claims confirm that the subset may comprise one candidate. Limitations [9e], [16e], and [18e] compare motion information of a potential candidate with motion information of candidates in the subset of limitations [9d], [16d], and [18d]; dependent claims 10 and 26 further specify that the potential candidate is compared with "at most one other" candidate. The specification includes embodiments where the subset is a single candidate. Ex-1001, 15:38-16:28 (e.g., block A1 is compared with block B1, block B0 is compared with block B1, block A0 is compared with block A1), Figs. 8b, 10a, 10b.

V. INVALIDITY GROUNDS

48. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSITA in the 2011 timeframe, and my training and experience, it is my opinion that a POSITA would have found the Challenged Claims of the '833 patent (as identified in the below chart) obvious in view of the invalidity grounds. The reasons for my conclusions are explained more fully below.

49. There are a number of patents and publications that constitute prior art to the ’833 patent. I have reviewed and considered the prior art discussed in this section, along with the materials listed in Appendix 2.

50. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSITA in the relevant timeframe (*supra* §II.C), and my training and experience, it is my opinion that the challenged claims of the ’833 patent are invalid based on the following grounds:

Grounds	Claims	Statutory Basis	Prior Art
1	9-14, 16, 18, 26-30	§ 103	Rusert and Zheng
2	9-14, 16, 18, 26-30	§ 103	Nakamura and WD4

51. I understand that Rusert and Zheng were not cited or considered during prosecution of the ’833 patent. WD4 was cited in an Information Disclosure Statement filed on September 25, 2015, but was never relied upon by the Examiner for a rejection. Ex-1002, 000266. In addition, I reviewed the file history for the ’833 patent and am not aware of these references being discussed in any office action or in the prosecution history generally. I understand that Nakamura was cited in an IDS that was signed after the examiner issued a Notice of Allowance for the ’833 patent. *Supra* §II.B. Additionally, it appears that only

the 10-page Nakamura document was specifically cited in the Information Disclosure statement submitted on April 25, 2016. Ex-1002, 000896.

A. Ground 1

52. It is my opinion that a POSITA would have found the Challenged Claims obvious based on the combination of Rusert and Zheng (Ground 1). Rusert provides various teachings for selecting motion vector predictor candidates for video encoding and decoding. Zheng provides additional teachings that update Rusert's teachings and terminology in accordance with the H.265 standard, including for example teachings regarding a type of block called a prediction unit or "PU" in H.265 and a list of candidates for motion prediction called a "merge list" or a "merge set." Their combination, including the motivation to combine and reasonable expectation of success, is explained in more detail in the claim analysis sections below. E.g., *infra* §V.A.3[9a].

1. Rusert (Ex-1004)

53. Rusert is U.S. Patent Application Publication No. 2011/0194609 to Rusert et al., entitled "Selecting Predicted Motion Vector Candidates." Rusert was filed on February 7, 2011, and was published on August 11, 2011. I understand Rusert is prior art to the '833 patents under at least pre-AIA §102(a) and §102(e).

54. I have reviewed the Rusert reference. I understand that Rusert was not cited or considered during prosecution of the '833 patent based primarily on

the fact that Rusert is not cited on the face of the ’833 patent and was not discussed in the prosecution history.

55. Rusert teaches a method for selecting Prediction Motion Vector candidates (“PMV candidates”) for video encoding and decoding. Ex-1004, Abstract. Rusert reduces the number of candidates using a “subset of the set of previously coded motion vectors that were used for previous blocks[,]” e.g., by limiting the candidates to those used “for previous blocks having an allowed distance from the current block.” Ex-1004, ¶11. When the list of candidates is updated, duplicates and essentially “similar” vectors are excluded. *Id.* ¶¶70-73. After the candidate list is completely formed, a candidate is selected from the list and used for motion prediction by the encoder that is then signaled to the decoder which independently creates the same candidate list. *Id.* ¶¶12-13, ¶¶88-99.

56. The list of PMV candidates is called “PMV_CANDS.” I note that Rusert includes some typographical errors where this name is mistyped as “PMV_SANDS.” E.g., Ex-1004, ¶6, ¶39, ¶42, ¶44, ¶88. This is clearly a typo, which would have been clear to POSITA in the context of Rusert. Several paragraphs use both “PMV_SANDS” and “PMV_CANDS” to refer to the same list. *See*, *e.g.*, Ex-1004, ¶39 (describing how “[u]pdate means that one or more motion vectors are added to an existing PMV_SANDS list[]” and how “[a] PMV_CANDS list may be updated”), ¶42 (referring to “one or multiple

PMV_CANDS lists" and "either a single or two different PMV_SAND"), ¶44 (describing how "[t]he PMV_CANDS list may be updated" and how "to update PMV_SANDS[]"), ¶88 (referring to "candidates in PMV_SANDS" and "candidates in the PMV_CANDS list."). Moreover, the places where Rusert refers to "PMV_SANDS" have corresponding discussions in Rusert's provisional application that correctly state "PMV_CANDS". *Compare* Ex-1004, ¶6 *with* Ex-1012, 000003-4. Furthermore, this name is easily mistyped because the letter "S" is close to "C" on QWERTY keyboards and both appear in the name. Moreover, this type of typographical error is commonplace with Microsoft Word because it is all capital letters, which Microsoft Word often skips (there is a setting for the spell check to omit words in all capitals).

57. Rusert explains its teachings in the context of H.264, and therefore Rusert's embodiments and the contextual background explanations of H.264 are part of the same teachings. Moreover, A POSITA would have been motivated to combine Rusert's teachings from its embodiments with Rusert's teachings regarding the operation of H.264 in its background section. Rusert explains that its teachings are given with examples in H.264 and therefore provides express teaching, suggestion, and motivation to combine the teachings of its embodiments with known concepts. *See, e.g.*, Ex-1004, ¶¶3-4,¶67, ¶77, ¶116.

58. Rusert is analogous art in the same field as the ’833 patent (video encoding and decoding, *infra* §IV). For example, Rusert teaches a method for a video encoding and decoding apparatus (Ex-1004, ¶1), with motion vector teachings used to encode and decode video (*id.*, ¶¶23-25).

2. Zheng (Ex-1005)

59. Zheng is U.S. Patent Application Publication No. 2013/0077691 to Zheng et al., entitled “Parallelization Friendly Merge Candidates for Video Coding.” Zheng was filed on June 18, 2012, and claims priority to six provisional applications:

- U.S. Provisional Application No. 61/499,112 filed on June 20, 2011 (Ex-1054, “Zheng ’112 Provisional”).
- U.S. Provisional Application No. 61/543,043 filed on October 4, 2011 (Ex-1055, “Zheng ’043 Provisional”).
- U.S. Provisional Application No. 61/543,059 filed on October 4, 2011 (Ex-1056, “Zheng ’059 Provisional”).
- U.S. Provisional Application No. 61/556,761 filed on November 7, 2011 (Ex-1057, “Zheng ’761 Provisional”).
- U.S. Provisional Application No. 61/562,387 filed on November 21, 2011 (Ex-1058, “Zheng ’387 Provisional”).

- U.S. Provisional Application No. 61/562,953 filed on November 22, 2011 (Ex-1059, "Zheng '953 Provisional").

I understand that Zheng is prior art to the '833 patent under at least pre-AIA §102(e).

60. Additionally, it is my understanding that a prior art patent is entitled to its provisional application date if the subject matter relied upon in the patent is described in the provisional application, and at least one of the claims of the published application is supported by the written description of the provisional application. It is my opinion that Zheng and the relied-upon disclosures are entitled to a priority date of October 4, 2011 (the filing date of Zheng '043 Provisional (Ex-1055) and Zheng '059 Provisional (Ex-1056)), which is before the earliest possible priority date of November 4, 2011, of the '833 patent. First, Zheng's first three provisional applications (Zheng '112 Provisional (Ex-1054), Zheng '043 Provisional (Ex-1055), and Zheng '059 Provisional (Ex-1056)) provide support for all subject matter relied-upon in challenging the validity of the Challenged Claims, and I have included cross-citations in my claim analysis sections of where the provisional applications support the relevant teachings from Zheng. Second, as set forth in the following table, the Zheng Provisionals provide sufficient details to satisfy the written description requirements of 35 U.S.C. § 112.

to make or use the invention claimed by Zheng, including as stated in Zheng's claim 1.

Zheng's Claim 1	**Zheng's Provisional Support**
1. A method of coding video data comprising:	Ex-1055, 000002 (Figure 1), 000004 (Figures 3A and 3B), 000005 (Figure 4), 000006 (Figure 5), 000015 (¶6), 000016 (¶17), 000035 (claim 1)
determining a merge candidate set for a current prediction unit of a current coding unit, wherein the merge candidate set is determined without comparing motion information of a merge candidate in the merge candidate set to motion information of any other prediction units; and	Ex-1055, 000015 (¶6), 000019-22 (¶¶32-43), 000035 (claim 1)
performing a merge motion vector prediction process for the current prediction unit using the merge candidate set.	Ex-1055, 000017-18 (¶¶25-27, 29), 000021-22 (¶¶41-44), 000027-28 (¶¶63-64)

61. Finally, none of Zheng's provisional applications that were filed after the earliest possible priority date of November 4, 2011, are relied upon for establishing written description support of Claim 1 of Zheng, and none of these disclosures are relied upon for demonstrating invalidity of the '833 patent.

62. I have reviewed the Zheng reference. I understand that Zheng was not cited nor considered during prosecution of the '833 patent based primarily on the fact that Zheng is not cited on the face of the patent nor discussed in the prosecution history.

63. Zheng provides video encoding/decoding teachings related to the High Efficiency Video Coding ("HEVC") video standard, also referred to as H.265. For example, Zheng teaches coding units (CUs) and prediction units (PUs), which are concepts in H.265 and correspond to macroblocks in H.264. Ex-1005, ¶¶28-29, 32, 38; Ex-1056, 000016 (¶¶17-18, 20), 000017 (¶¶22-23), 000018 (¶25). Zheng also teaches creating a "merge candidate list" for selecting merge candidates for a block. Ex-1005, ¶¶25, 45, 47, 49-56; Ex-1055, 000016 (¶17), 000018-19 (¶¶28-31), 000019-20 (¶¶33-36).

64. Zheng also teaches removing redundant motion vector prediction candidates in merge mode when a coding unit is subdivided into two or more prediction units. Ex-1005, ¶29; Ex-1055, 000016 (¶20). When a coding unit is divided into two prediction units, Zheng teaches that the candidate from the first prediction unit can be removed because it is redundant. Ex-1005, ¶49; Ex-1055, 000019 (¶33).

65. Zheng is analogous art in the same field as the '833 patent (video encoding and decoding, *infra* §V). For example, Zheng teaches that its techniques relate to "video coding" (Ex-1005, Title, ¶2; Ex-1055, 000014 (¶1)) and that its disclosed process "may also reduce the computational complexity of encoding and decoding." Ex-1005, ¶ 63; Ex-1055, 000021 (¶38); *see also* Ex-1005, ¶¶86, 152; Ex-1055, 000022 (¶45), 000033 (¶88).

3. Independent Claim 9

[9pre] A method comprising:

66. To the extent that the preamble is limiting, Ground 1 teaches limitation [9pre], as explained below.

67. Rusert teaches a **method**. For example, Rusert teaches "a *method* of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from a current block." Ex-1004, Abstract, ¶1, ¶11. The method further comprises the steps and elements explained below. *Infra* §§V.A.3[9a]-[9g].

[9a] receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;

68. Ground 1 teaches limitation [9a], as explained below.

69. Rusert teaches **receiving an encoded block of pixels including a prediction unit.** Rusert teaches "receiv[ing] the transmitted encoded video stream[.]" Ex-1004, ¶34, Fig. 1:

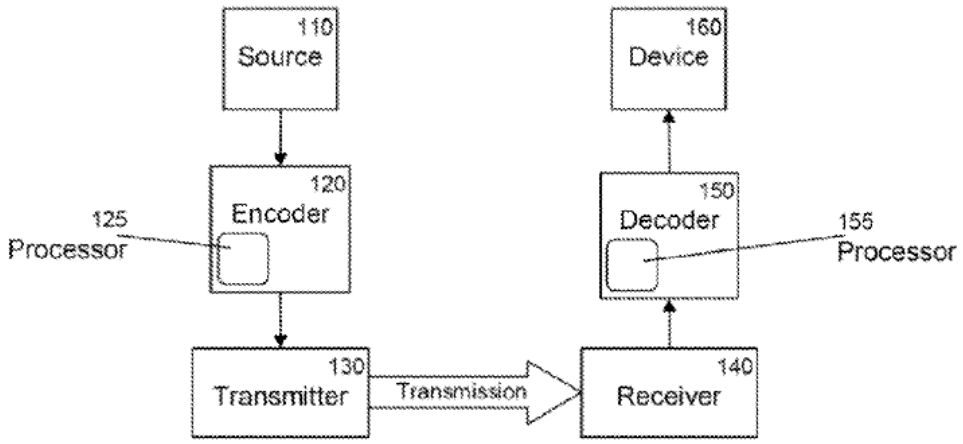


Fig. 1

This video stream comprises encoded "pixel blocks" received from the encoder. Ex-1004, ¶2, ¶36 ("e.g. 8x8 pixel block… 16x16 block"). Rusert teaches that the "encoder 120 applies an encoding process to the video signal to create an encoded video stream" that is then transmitted to the receiver that "receives the transmitted encoded video stream and passes this to a decoder 150." Ex-1004, ¶34.

70. Rusert explains its teachings with references to blocks to which motion vectors are assigned. Ex-1004, ¶¶2-5, ¶36, ¶43. Therefore, in the context of Rusert's nomenclature, the prediction unit included in the block of pixels is the block itself because the block is the unit for which a motion vector is assigned for motion prediction. Ex-1004, ¶2 ("Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames)."), ¶3 ("in recent video coding standards such as H.264/AVC where *small motion*

compensation block sizes are used"), ¶4, ¶36, ¶43 ("Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks.").

71. Additionally, the combination of Ground 1 applies Rusert's teachings regarding "blocks" (in the H.264 context) to prediction units (following nomenclature in the H.265 context), as taught by Zheng. As further explained below, Rusert explains its teachings in terms of blocks, with motion vectors assigned to or obtained from blocks in the H.264 context. Ex-1004, ¶¶2-5, ¶36, ¶43. Zheng explains that, for H.265, a PU is a type of block, and motion vectors are assigned to PUs. Ex-1005, ¶¶3, 27-29, 38; Ex-1055, 000014 (¶2), 000016-17 (¶¶19-22), 000017 (¶24). For example, Zheng uses the term "current block" to reference a current prediction unit. Ex-1005, ¶ 38 ("Coding a PU using inter-prediction involves calculating a motion vector between a *current block* and a block in a reference frame."); Ex-1055, 000017 (¶24). Therefore, based on these combined teachings, it would have been obvious to apply Rusert's motion vector teachings to PUs in the H.265 context taught by Zheng, and the combination teaches receiving an encoded block of pixels including a prediction unit. E.g., Ex-1005, ¶107 ("The motion estimation unit 42 calculates a motion vector for a prediction unit of an inter-coded frame by comparing the prediction unit to

reference samples of a reference frame stored in the reference frame buffer 64."); Ex-1055, 000027 (¶63); *see also* Ex-1005, ¶¶33-36, ¶66; Ex-1055, 000016-17 (¶22), 000031 (¶79), 000032 (¶84).

72. **Motivation to Combine Zheng's PU Teachings with Rusert**. Rusert uses H.264 terminology and teaches the blocks that serve as prediction units in H.264. To the extent the claims require H.265 PUs, those were well-known in the art and for example taught by Zheng. A POSITA would have been motivated to apply Rusert's teachings to H.265 PUs and related concepts, as explained below.

73. Rusert provides express motivation to combine its teachings with other video standards. Rusert explains that, while its teachings and embodiments were "given in the context of H.264/AVC, *the principles disclosed herein can also be applied to … other coding standard[s], and indeed any coding system which uses predicted motion vectors*." Ex-1004, ¶116.

74. H.265 was the successor video standard to H.264. Ex-1011, ¶¶32-36. By 2011, it had been widely known that H.265 was emerging as the successor video standard. *Id.* ("[t]he emerging HEVC standard … referred to as ITU-T H.265[.]"). Both H.264 and H.265 standards were drafted by ITU-T, the Telecommunication Standardization Sector of the International Telecommunication Union. H.265 used H.264 as a baseline starting point and used the same concepts of block-based video encoding with predicted motion vectors

for those blocks. Ex-1011, ¶35 ("[T]he PU may include data defining a motion vector for the PU."), ¶37 ("Each of the blocks may be predictively encoded based on blocks of previously coded pixels[.]"), ¶66, ¶71; Ex-1005, ¶3 ("Digital video devices implement video compression techniques, such as those described in the standards defined by MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Part 10, Advanced Video Coding (AVC), the High Efficiency Video Coding (HEVC) standard presently under development, and extensions of such standards, to transmit, receive and store digital video information more efficiently."; Ex-1055, 000014 (¶2). Therefore, H.265 is a coding standard that uses predicted motion vectors—precisely the type of coding standard for which Rusert provides express motivation to combine.

75. Rusert's motion-vector teachings are explained on the basis of "blocks" because Rusert applies H.264 terminology, where motion vectors are assigned to "blocks." Ex-1004, ¶¶2-5, ¶36, ¶43. As Zheng explains, in H.265, a block can be a "prediction units" ("PUs"), which are assigned motion vectors to perform predictions. Ex-1005, ¶29, ¶32 ("This disclosure also uses the term 'block' or 'portion' to refer to any of a CU, PU, or TU."), ¶36 ("To code a block (e.g., a prediction unit (PU) of video data), a predictor for the block is first derived."); Ex-1055, 000016 (¶20), 000016-17 (¶22), 000017 (¶24); *supra* §I. For example, Zheng teaches that "[c]oding a PU using inter-prediction involves

calculating a motion vector between a current block and a block in a reference frame. … A motion vector, for example, may indicate the displacement of a prediction unit in a current frame relative to a reference sample of a reference frame." Ex-1005, ¶38; Ex-1055, 000017 (¶24).

76. While Rusert's express teaching, suggestion, or motivation was sufficient, on its own, a POSITA would have been motivated to make this combination for a number of additional reasons, explained below.

77. For example, a POSITA would have been motivated to combine Rusert and Zheng because it would have combined prior art elements according to known methods to yield predictable results. For example, the combination would have combined Rusert's teachings for generating/de-duplicating PMV candidate lists, including for the selection of PMV candidates for blocks, with H.265 concepts (e.g., prediction units and related information for motion vectors), according to known methods (e.g., as taught by Zheng and known throughout the industry based on the widespread knowledge of H.265). Rusert explains its teachings in terms of blocks, with motion vectors and motion prediction operating in terms of those blocks. Ex-1004, ¶¶2-5, ¶36, ¶43. H.265 drafts introduced new terminology for a type of block called a "prediction unit" ("PU"), with motion vectors and motion prediction operating in terms of PUs, where PUs comprised the blocks that are used for motion prediction. Ex-1011, ¶¶33-36, ¶¶64-66. These

terminology and concepts are adopted in Zheng. Ex-1005, ¶29, ¶32, ¶36; Ex-1055, 000016 (¶20), 000016-17 (¶22), 000017 (¶24). For example, Zheng teaches that "… a coding unit may have variable sizes depending on video content. … a coding unit may be split into smaller blocks for prediction or transform. In particular, each coding unit may be further partitioned into prediction units (PUs) and transform units (TUs)." Ex-1005, ¶29; Ex-1055, 000016 (¶20). In addition, Zheng teaches that motion information for prediction used in video coding includes motion vectors. Ex-1005, ¶40, ¶49; Ex-1055, 000017 (¶25), 000019 (¶33). Therefore, it would have been obvious to apply Rusert's motion-vector teachings to PUs in the H.265 context as a combination of prior art elements according to known methods, with predictable results as explained further below.

78. The combination would also have used Rusert's known techniques (e.g., for efficiently selecting PMV candidates) to improve similar devices and methods (e.g., for H.264 and H.265) in the same way. Rusert provides block-based motion vector teachings that were explained using H.264 as an example but were applicable to a variety of video coding standards. Ex-1004, ¶2, ¶116. This would have included the H.265 standard, which was similar to and in fact derived from H.264. Ex-1011, ¶3, ¶32 ("[T]he HEVC Test Model… presumes several capabilities of video coding devices over devices according to, e.g., ITU-T H.264/AVC."); *supra* §II.D. Zheng teaches aspects of the H.265 standard, such as

PUs and motion vector prediction. Like Rusert, Zheng teaches a block-based video coding approach. Ex-1005, ¶4 ("For block-based video coding, …"), ¶36, ¶40; Ex-1055, 000014 (¶3), 000016-17 (¶22). Therefore, it would have been natural to apply Rusert's teachings regarding blocks to the H.265 context, including by applying Rusert's teachings to PUs as taught by Zheng, to improve the similar H.265 standard in the same way that Rusert explained for H.264. This would have applied a known technique to a known device/method that was ready for improvement, to yield predictable results as further explained below.

79. Furthermore, the combination of Rusert and Zheng was a simple substitution of a known element (e.g., Zheng's prediction unit teachings) into Rusert's teachings. Rusert explains its teachings in terms of the blocks that form the basis for motion vector operations, and Zheng explains that, for H.265 video, those blocks (whose sizes are now allowed to vary) were called "PUs." Ex-1005, ¶¶28-29; Ex-1055, 000016 (¶¶19-22). Therefore, a POSITA would have been motivated to combine Rusert and Zheng by applying Zheng's teachings regarding PUs into Rusert's teachings for selecting PMV candidates for blocks in video frames. A POSITA would have understood the features introduced by H.265 do not prevent teachings with respect to H.264 from being applied to H.265, and vice-versa. Effectively, Zheng provided the ability for the encoded block using motion

prediction to be based on different block sizes, which it called a PU. Ex-1005, ¶¶36-38, 40; Ex-1055, 000016-17 (¶¶22-25).

80. Additionally, a POSITA would have found motivation in the similarity of the references. Both Rusert and Zheng are directed to video coding. Ex-1004, ¶1 ("The present application relates to … a video encoding apparatus, a video decoding apparatus …"); Ex-1005, Abstract ("This disclosure presents methods and systems for coding video in merge mode of a motion vector prediction process."); Ex-1055, 000037 (Abstract). Both are discussed in the context of ITU standards, including H.264 and H.265. Ex-1004, ¶¶3-4, ¶67, ¶116 ("specific examples have been given in the context of H.264/AVC"); Ex-1005, ¶3 (… video compression techniques, such as those described in the standards defined by MPEG-2, MPEG-4, ITU-T H.263, **ITU-T H.264**/MPEG-4, Part 10, Advanced Video Coding (AVC) the **High Efficiency Video Coding (HEVC)** standard presently under development …"), ¶¶27-28; Ex-1055, 000014 (¶2), 000016 (¶19). And while H.265 included new terminology, e.g., for prediction units ("PU"), Zheng explains that this unit is a block of pixels. Ex-1005, ¶29 (describing that coding units may be partitioned into "smaller blocks" such as prediction units and transform units); Ex-1055, 000016 (¶20). Zheng further explains that its coding unit "may be considered to be similar to a so-called macroblock, e.g., under other

video coding standards such as ITU-T H.264. Ex-1005, ¶28; Ex-1055, 000016 (¶19).

81. A POSITA would have been motivated because the combination of Rusert and Zheng would have yielded several advantages. For example, Rusert provides for "an improved method and apparatus for selecting PMV candidates" "to improve video coding efficiency[.]" Ex-1004, ¶7. Zheng provides teachings "[t]o achieve better coding efficiency." Ex-1005, ¶29, ¶¶39-40; Ex-1055, 000016 (¶20), 000017 (¶25).

82. Additionally, a POSITA would have understood that Zheng's teachings regarding partitioning of coding units into prediction units and transform units are applicable to both H.264 and H.265. *See* Ex-1005, ¶4 (describing a block-based video coding approach), ¶29 ("Prediction units may be considered to be similar to so-called partitions under other video coding standards, such as H.264."; Ex-1055, 000014 (¶3), 000016 (¶20). Rusert's teachings and Zheng's teachings are applicable to coding and decoding blocks of pixels, regardless of the nomenclature used to label them.

83. Moreover, the combination would have had predictable results. Rusert already applies its teachings to block-based video encoding/decoding—i.e., where the operation of motion vectors is based on blocks. *See, e.g.*, Ex-1004, ¶2, ¶11. Zheng explains that a PU is a type of block; a POSITA would have

understood that the operation of motion vectors in H.265 is based on PUs. Ex-1005, ¶29, ¶32, ¶36; Ex-1055, 000016 (¶20), 000016-17 (¶22), 000017 (¶24); *see also* Ex-1011, ¶35. Therefore, the concepts taught in Rusert were readily applicable to PUs, and the combination would have had the predictable result of selecting PMV candidates (as Rusert teaches) for PUs (as Zheng teaches). *See, e.g.*, Ex-1004, ¶116.

84. Furthermore, the combination would not have changed the principle of operation for any of the teachings of the references relied upon in the combination. The combination applies teachings from Rusert and Zheng in the manner taught by each reference: Rusert's teachings are still applied to the selected PMV candidates for block-based video encoding/decoding using predicted motion vectors, and Zheng's teachings are still used with motion vectors assigned to PUs, as explained above. Applying H.265 teachings to Rusert, and vice versa, would have been consistent with Rusert's statement that its principles are applicable to other video standards. *Id*.

85. A POSITA would have had a reasonable expectation of success when combining Rusert and Zheng. As explained above, the combination applies teachings from each reference according to their known purposes, in a conventional manner as taught by Rusert and Zheng, without changing their principle of operation. Furthermore, the teachings from Rusert and Zheng are

complementary because both teach aspects of block-based video encoding from H.264 and H.265, respectively. Rusert provides examples using H.264, explaining it was not limited to that standard and its teachings apply to other coding standards, which naturally would have included the successor standard H.265. *Id*. Therefore, Zheng's teachings complement Rusert by teaching terminology and concepts from H.265. *See, e.g.,* Ex-1005, ¶29, ¶32, ¶36; Ex-1055, 000016 (¶20), 000016-17 (¶22), 000017 (¶24); *see also* Ex-1011, ¶¶32-35.

86. Additionally, the combination had predictable results, as explained above, and Ground 1 does not modify Rusert or Zheng in a way that would render either reference inoperative. To the contrary, the similarities of the architectures and video standards—which both use block-based video encoding/decoding with predicted motion vectors assigned to blocks—would have given a POSITA a reasonable expectation of success in combining their teachings. *See, e.g.*, Ex-1004, ¶11 (discussing motion vectors used for coding blocks); Ex-1005, ¶¶4-5 (providing an overview of block-based video coding); Ex-1055, 000014-15 (¶¶3-4).

87. Given the education and experience of a POSITA (*supra* §III), a POSITA would have been more than capable of applying Zheng's teachings to Rusert, and vice versa, because it would simply have applied Rusert's teachings from H.264 to the successor standard H.265. Additionally, motion estimation was commonplace, having been introduced by MPEG standards in the 1990s, and the

use of predicted motion vectors was standardized in H.264 by the early-mid 2000s. *Supra* §I. These were basic aspects of video encoding/decoding that a POSITA would have been knowledgeable about. In essence, the combination of Rusert and Zheng is the application of block-based motion vector prediction to the variable sized blocks of H.265 with the new nomenclature of CU, TU, and PU.

88. In summary, for the reasons explained above, a POSITA would have found it obvious and been motivated to combine Rusert's block-based teachings with Zheng's H.265 teachings, including its teachings of prediction units.

89. Ground 1 further teaches **receiving … information** (e.g., an index code) **identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder**. Rusert teaches receiving "a code 'index'" sent "from the encoder… to the decoder" that identifies "a particular PMV candidate … from a list of PMV candidates, PMV_CANDS[.]" Ex-1004, ¶36, ¶37 ("Using the transmitted index, the decoder… can determine the PMV… as used in the encoder… and thus may reconstruct MV=DMV+PMV."), Fig. 1:

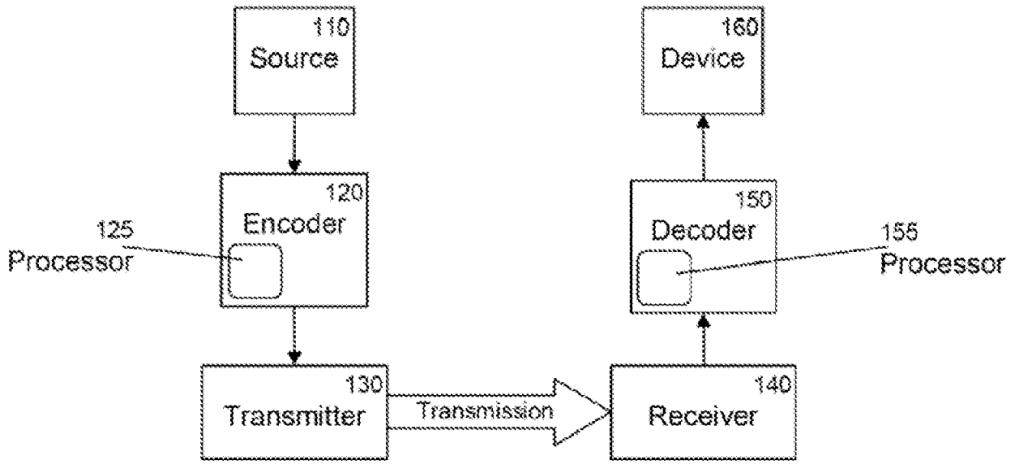


Fig. 1

90. The index is information received by Rusert's decoder to "select a particular PMV candidate… from a list of PMV candidates, PMV_CANDS" and the selected PMV candidate is used to reconstruct the motion vector for the current block. Ex-1004, ¶¶36-37. In this way, the received index is an index into the merge list by pointing to a particular candidate based on its position in the merge list. Rusert teaches zero-based indexing examples, starting with zero for the first candidate and ending with n-1 for a list of n candidates. Ex-1004, ¶¶88-95. PMV_CANDS is constructed by an encoder, and the decoder "mimics the encoder" to construct the same PMV_CANDS list. Ex-1004, ¶35, ¶37, ¶39. From PMV_CANDS, the decoder uses the index "signaled from the encoder" to "select a particular PMV candidate[.]" Ex-1004, ¶36. "Using the transmitted index, the decoder… can determine the PMV 220" to "reconstruct [the motion vector.]" Ex-1004, ¶37.

91. I note that Rusert teaches two options for initializing and updating PMV_CANDS: (1) PMV_CANDS is "dynamically generated specifically for the current motion compensation block" and (2) PMV_CANDS is "initialized once" and "updated according to a sliding window approach." Ex-1004, ¶41. A POSITA would have been motivated to use the first option in which PMV_CANDS is initialized and updated dynamically for each block because Rusert presents it as the first option. Furthermore, the first option is one of only two options, and a POSITA would have been motivated to apply at least the first option as one of a finite number of options that Rusert teaches. For example, when Rusert's PMV_CANDS list is dynamically constructed, after it is complete, it is a merge list.

92. Rusert stores its motion vector prediction candidates in a list called PMV_CANDS. Ex-1004, ¶¶36-¶39. Zheng teaches that, for H.265 merge mode, candidates for motion prediction are stored in a list called the "merge candidate set." Ex-1005, ¶47 ("Merge candidates are generated per PU. That is, each PU has its own set of merge candidates."); Ex-1055, 000018 (¶29); *see also* Ex-1005, ¶50 ("… HEVC proposes comparing the motion information for each merge candidate in a merge candidate set with the motion information of other PUs of the same CU. Any merge candidates with the same motion information as a previously coded PU is removed from the merge candidate set to avoid the whole CU using the same

motion information."); Ex-1055, 000019 (¶33); Ex-1005, ¶¶51-56 (discussing the process on how to generate a merge candidate set for a PU); Ex-1055, 000019 (¶35). A merge candidate set and a merge list refer to the same concept in the context of generating prediction candidates. A POSITA would have understood this because these terms were used interchangeably in the art to refer to the candidates that were used for motion prediction. Ex-1005, Abstract; Ex-1055, 000037 (Abstract); Ex-1060, 000003. Zheng describes its construction of the merge candidate set as follows:

> 1. Check the motion information for the next candidate block.
>
> 2. Compare the candidate motion information with the motion information for previously coded PUs in the same CU
>
> 3. If the candidate motion information for the candidate block is the same as the motion information for a previously coded PU, go to step 1; otherwise, go to step 4
>
> 4. Add the candidate block into the merge candidate set
>
> 5. If all the candidate blocks are checked, finish; otherwise, go to step 1

Ex-1005, ¶¶51-56; Ex-1055, 000019 (¶35). In both Rusert and Zheng, the video encoder and decoder independently construct the same list with the same candidates, regardless of what the list is called. Ex-1005, ¶86 ("FIG. 5 is a block diagram illustrating an example video encoding and decoding system 10 that may

be configured to utilize techniques for generating candidate motion vectors in a merge mode in accordance with examples of this disclosure."), Figs. 5-7; Ex-1055, 000022 (¶45), 000006-8 (Figs. 5-7); Ex-1004, ¶35, ¶¶24-25 ("[T]he encoder and decoder follow the same rules for creating and modifying the set of PMV candidates" "to "maintain synchronization" of the respective lists of candidates"). The encoder and decoder both construct the merge list to avoid the need to transmit the merge list from encoder to decoder, thereby reducing the size of the encoded video and any subsequent transmission or storage. *Id.*

93. Consistent with Rusert and confirming that their combined teachings would have operated in a similar manner, Zheng teaches receiving information identifying a spatial motion vector prediction candidate (e.g., a merge candidate) from its merge candidate list. For example, Zheng teaches that its "motion estimation unit 42 calculates a motion vector for a prediction unit of an inter-coded frame by comparing the prediction unit to reference samples of a reference frame stored in the reference frame buffer 64. … In merge mode, the encoder considers a set of merge candidate blocks and selects a block that has the same (or most closely matches) motion vector, reference frame, and prediction direction as the current block to be coded. … [R]ather than signaling this motion vector information (i.e., the motion vector, reference frame, and prediction direction) in the encoded bit stream, the encoder signals an *index number* for the selected

motion vector candidate." Ex-1005, ¶¶107, 109; Ex-1055, 000027 (¶63-64). Like Rusert, Zheng teaches identifying the best candidate block for the merge mode using an index. Ex-1005, ¶109 ("[T]he encoder signals an index number for the selected motion vector candidate. The index number identifies the selected candidate motion vector among a set of candidate motion vectors. A decoder may copy the motion vector information from the motion vector candidate to use for a current block."); Ex-1055, 000018 (¶27), 000027 (¶64). In this respect, MVP and merge modes operate in a similar fashion, where the decoder receives information that identifies which candidate in the list was chosen and signaled by the encoder to be used for motion prediction of the current block.

94. The candidates stored in Zheng's merge candidate set contain spatial motion vector prediction candidates because the spatial candidates represent a candidate motion vector obtained from one or more previously-encoded blocks (e.g., the neighbors above and to the left) in the current frame that can be used to perform motion compensation at the decoder. Ex-1005, ¶50 ("HEVC proposes comparing the motion information for each merge candidate in a merge candidate set with the motion information of other PUs of the same CU. Any merge candidates with the same motion information as a previously coded PU is removed from the merge candidate set to avoid the whole CU using same motion information."); Ex-1055, 000019 (¶34). Zheng's index is an index into the merge

candidate set (a merge list) that identifies either a spatial or temporal motion vector prediction candidate from the merge candidate set constructed by an encoder that will be used for the current block.

95. Zheng's merge mode reduces the signaling from the encoder to the decoder, by having the encoder "instruct[] a decoder, through bitstream signaling of prediction syntax, to *copy a motion vector*, reference index (identifying a reference frame, in a given reference picture list, to which the motion vector points) and the motion prediction direction (which identifies the reference picture list (List 0 or List 1), i.e., in terms of whether the reference frame temporally precedes or follows the currently frame) from a selected candidate motion vector for a current portion of the frame" Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27). Thus, in some instances there will be no need to send a residual signal as the "candidate motion vector will have already been decoded by the decoder. … the decoder may simply retrieve the motion vector, reference index, and motion prediction direction associated with the causal PU from memory and copy these values for the current PU." Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27). By using the previously decoded blocks for both pixel values and motion information, the encoder does not need to signal a motion difference vector or a residual difference signal, thereby reducing signaling from the encoder to the decoder, thus making the bitstream smaller.

96. **Motivation to Apply Zheng's Merge Candidate Set Teachings.** A POSITA would have been motivated to combine teachings from Rusert and Zheng, with Rusert's PMV_CANDS used as a merge list for a block of pixels that includes a PU. Ex-1005, ¶¶45, 47; Ex-1055, 18 (¶¶28-29). This would have been a simple application of Rusert's teachings in an H.265 context. As explained above, Rusert provided express motivation to combine its teachings with other standards beyond H.264. Ex-1004, ¶116; *supra* ¶¶72-74.

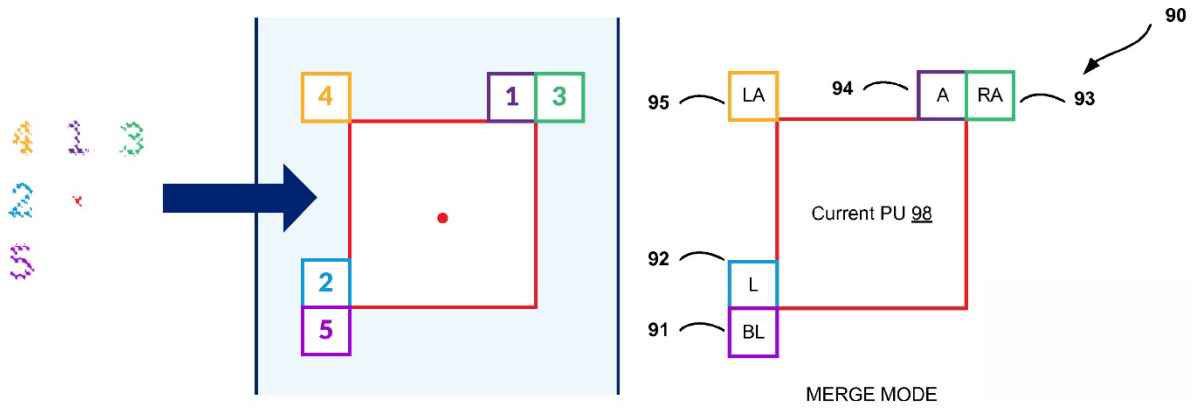
97. H.265 included a mode of operation called "merge mode," which was optimized for areas of uniform motion. Generally, under H.265, merge mode saved bits by utilizing the predicted motion vector without signaling difference vectors. As Zheng explains, "for merge mode, the prediction syntax may include a flag identifying the mode (in this case 'merge' mode) and an index identifying the selected candidate motion vector. … the candidate motion vector will have already been decoded by the decoder. As such, the decoder has already received and/or determined the motion vector, reference index, and motion prediction direction for the causal PU." Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27). For example, "the decoder may simply retrieve the motion vector, reference index, and motion prediction direction associated with the causal PU from memory and copy these values for the current PU" which eliminates the need of signaling a difference vector between the current block and the reference block. Ex-1005, ¶41; Ex-1055,

000017-18 (¶¶26-27); *see also* Ex-1006, ¶7 ("When a PU is coded in … Merge mode, no motion information is transmitted except for the index of the selected candidate.").

98. Zheng teaches merge mode, and a POSITA would have been motivated for a number of reasons to apply Zheng's merge mode teachings to Rusert, with PMV_CANDS used as a merge list. First Rusert is expressly directed at encoding efficiency and reducing signaling overhead. Rusert identifies that "the higher number of candidates, the higher number of bits required for signaling one of the candidates, which in turn causes additional overhead and thus reduced compression efficiency." Ex-1004, ¶6. Rusert noted that "an improved method" to "improve video coding efficiency" is required. Ex-1004, ¶7. Zheng's merge mode provides additional reduction in signaling, by eliminating Rusert's difference motion vector.

99. Second, Rusert teaches a pattern that corresponds to Zheng's pattern for merge mode, where candidates are evaluated from the five blocks that are immediate neighbors of the block currently being encoded. Rusert teaches examples where PMV_CANDS is limited to five or fewer, including an embodiment that uses "a maximum of four candidates in the list." Ex-1004, ¶107. When PMV_CANDS is limited to a maximum of five or fewer candidates, Rusert's preferred scan pattern ("[t]he search pattern shown in FIG. 3n") comprises

the immediately neighboring blocks in the row above and column to the left of the current block as illustrated in Figure 3n (Ex-1004, ¶¶65-66, ¶107, Fig. 3n), which teaches that "candidate motion vector set 90 currently proposed in the HEVC standard for use in merge mode. Merge mode uses six merge candidates from the following spatial and temporal blocks: the below left (BL) block 91, the left (L) block 92, the right above (RA) block 93, the above (A) block 94, the left above (LA) block 95, and the temporal block (T) 96." Ex-1005, ¶45, Fig. 1; Ex-1055, 000002 (Fig. 1), 000018 (¶28). Since encoding and decoding generally progresses from the top-left block to the bottom-right block, when the current block is being evaluated, motion information will only have been assigned to the neighboring blocks from the rows above and columns to the left of the current block.



Rusert, Fig. 3n Scan Order	Rusert Fig. 3n Reformatted	Zheng, Fig. 1

Rusert even suggests that Figure 3n's scan pattern "has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance" and when limited to the first 5 candidates, matches Zheng's search pattern. Ex-1004, ¶66. It would have been obvious to use Rusert's pattern with all of the "adjacent" neighboring spatial blocks used in the scan pattern, resulting in 5 candidate spatial blocks. This corresponds with Zheng's search pattern.

100. Therefore, in view of the similarities between these patterns, a POSITA would have found it obvious to apply Zheng's merge list teachings to Rusert's overall algorithm to reduce overall signaling through the use of Zheng's merge mode. *See, e.g.,* Ex-1004, ¶65 ("[A]ll blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded."); Ex-1005, ¶45 ("Merge mode uses six merge candidates from the following spatial and temporal blocks: …"); Ex-1055, 000018 (¶28). Rusert explains that limiting the search in this manner is advantageous because it "reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding." Ex-1004, ¶12.

101. Zheng provides additional motivation to combine by teaching the advantages of using merge mode. For example, Zheng teaches that "signaling the original motion vector directly may result in less efficient coding, as a large

number of bits are typically needed to convey the information." Ex-1005, ¶39; Ex-1055, 000017 (¶25). Accordingly, Zheng aims to reduce the amount of information signaling, thereby increasing the compression ratio and improving the efficiency of the encoding/decoding scheme. Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27); *supra* ¶95.

102. Moreover, the prior art provides express motivation to apply Rusert's teachings with a maximum of 5 or fewer candidates, which as explained above results in a scan pattern that matches H.265 merge mode, because it was known in the art that in background portions of the frame or scenarios where objects moved across the screen, neighboring blocks often had the same motion information. Ex-1005, ¶¶49-50 (teaching the advantage of removing redundant candidates with the same motion information); Ex-1055, 000019 (¶¶33-34); Ex-1005, ¶69 (explaining that including redundant candidates in the merge candidate set decreases performance due to extra bits that are required for signaling); Ex-1055, 000021 (¶41); Ex-1007, Fig. 1 ("Remove candidates with the same motion information"), Table 3; Ex-1009, 000013. For example, a scene with a plain white wall has a large region of redundant white blocks, and it was known in the art that merge mode was designed to save space when encoding those types of redundant blocks, where the motion information and pixel values for each block matches its immediate neighbors. *See* Ex-1005, ¶41 ("In merge mode, a video encoder

instructs a decoder, through bitstream signaling of prediction syntax, to copy a motion vector, …"); Ex-1055, 000017-18 (¶¶26-27); Ex-1020, 000002; Ex-1060, 000002 ("Merging is particularly well-suited for regions of homogeneous motion that can be sufficiently described by using a single motion parameter set."). These types of scenes and images were common in video, for example in the background of movies, television shows, and other video content, which would benefit from the use of merge mode. Therefore, a POSITA would have been motivated to apply Zheng's merge mode teachings to Rusert to reduce the bitstream when encoding redundant blocks.

103. Furthermore, the combination would not have changed the principle of operation for either Rusert or Zheng. Rusert's teachings are still applied to the selected candidates for block-based video encoding/decoding using predicted motion vectors, while Zheng's teachings are still used with motion vectors assigned to PUs, as explained above. Applying Zheng's teachings on concepts and terminology in H.265 to Rusert, or vice versa, would have been consistent with Rusert's statement that its principles are applicable to other video standards. Ex-1004, ¶116.

104. A POSITA would have had a reasonable expectation of success when combining Rusert and Zheng. As I explained above, the combination applies teachings from Rusert and Zheng in the manner taught by each reference, without

changing their principle of operation. Furthermore, the teachings from Rusert and Zheng are complementary because both teach aspects of block-based video encoding. Rusert provides examples using H.264, explaining it was not limited to that standard and its teachings apply to other coding standards, which naturally would have included the successor standard H.265. *Id*. Therefore, Zheng's teachings complement Rusert by applying terminology and concepts from H.265, including the use of merge mode. Ex-1005, ¶¶28-29, 40-41; Ex-1055, 000014 (¶2), 000016 (¶¶20-21), 000017-18 (¶¶25-26).

105. Additionally, the combination would yield predictable results, as explained above. Ground 1 does not modify Rusert or Zheng in a way that would render either reference inoperative. To the contrary, the similarities of the architectures and video standards—both of which use block-based video encoding/decoding with predicted motion vectors assigned to blocks—would have given a POSITA a reasonable expectation of success in combining their teachings. *See, e.g.*, Ex-1004, ¶11 (discussing motion vectors used for coding blocks); Ex-1005, ¶4 ("For block-based video coding, a video frame or slice may be partitioned into blocks"); Ex-1055, 000014 (¶3).

106. **Encoders and Decoders**. I further note that both Zheng and Rusert teach that encoding and decoding are reciprocal (i.e., decoding is the inverse operation of encoding), which confirms that the above-discussed teachings are

applicable to both encoding and decoding. Ex-1005, ¶93 ("Each of the video encoder 20 and the video decoder 30 may form part of a respective encoder-decoder (CODEC) that is capable of encoding or decoding video data."); Ex-1055, 000024 (¶52); Ex-1004, ¶35, ¶¶24-25. Since the encoder and decoder independently construct the same list of candidates, a decoder using PMV_CANDS for decoding an encoded block of pixels would "follow the same rules" as the encoder that encoded the block of pixels to "maintain synchronization" of "the respective lists of PMV candidates stored in the encoder and decoder[.]" Ex-1004, ¶35, ¶¶24-25. Applying Zheng's merge list teachings would not have altered this approach because, for H.265, the encoder and decoder still construct the same list of candidates independently (Ex-1005, ¶¶99, 126; Ex-1055, 000025-26 (¶58), 000030 (¶75)) and any differences between the amount of information transmitted does not impact the process for building candidate lists and selecting a candidate for use with motion prediction of a block.

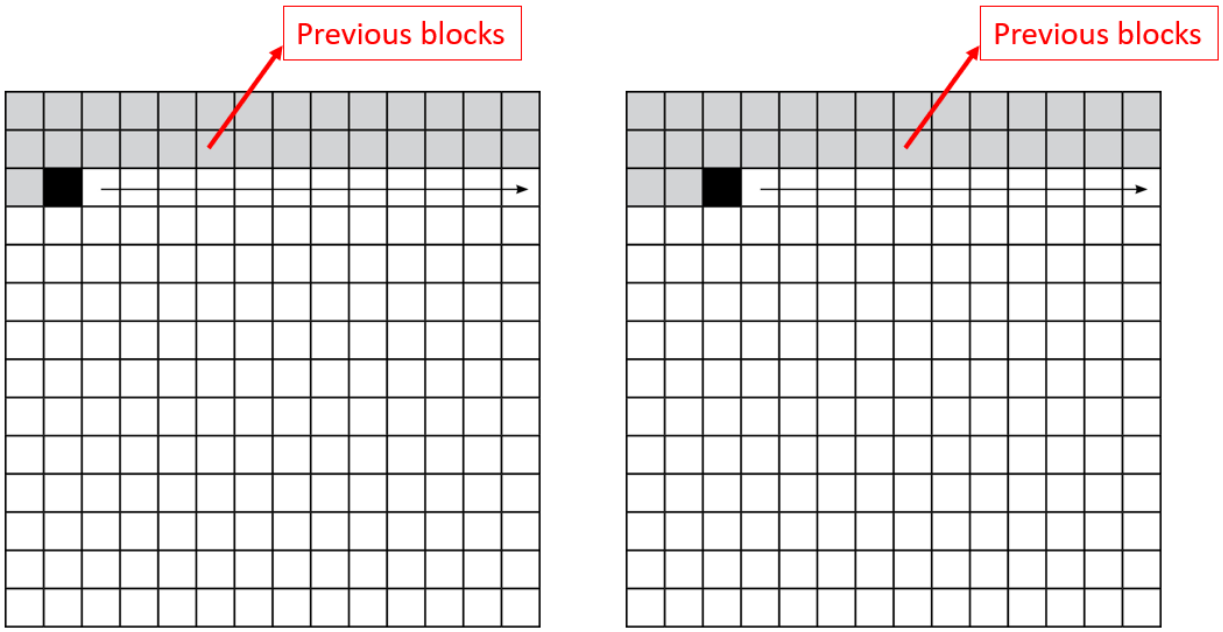
107. Based on the foregoing explanations, Ground 1 teaches limitation [9a].

[9b]	**determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

108. Ground 1 teaches limitation [9b], as explained below.

109. Ground 1 teaches **determining a set of spatial motion vector prediction candidates** (e.g., the set of previously-coded motion vectors) **for the encoded block of pixels.** Specifically, Rusert teaches a "set of previously coded motion vectors that were used for previous blocks … from the current block." Ex-1004, ¶11, ¶15, ¶¶24-25. In other words, in Rusert, for a given current block of pixels numbered block N (the Nth block of the frame), the set of spatial motion vector prediction candidates includes all previously-encoded blocks from 1 up to N-1, e.g.: 1, 2, 3, … N-2, N-1. For example, if we are currently encoding block 65, then the set of spatial motion vector prediction candidates for block 65 are blocks 1 through 64. Rusert teaches evaluating these previously-coded motion vectors as PMV candidates for potential inclusion in PMV_CANDS, based on distance from the current block, and these include spatial and temporal motion vector prediction candidates, e.g., "[m]otion vectors" that "comprise *spatial* or temporal *neighbors of the current block*[.]" Ex-1004, ¶67, ¶11, ¶12, ¶15, ¶¶24-25, ¶39, ¶113, Fig. 6. Since neighboring blocks are near the current block in the current frame, the PMV candidates from those neighboring blocks are obtained only on the basis of information of one or more blocks of the current frame. Ex-1004, ¶¶4-6. Thus, the previously-coded motion vectors are spatial motion vector prediction candidates that are obtained from one or more previously-decoded blocks in the current frame. *Supra* §IV.A.

110. The "set of previously coded motion vectors that were used for previous blocks[,]" is a set of spatial motion vector prediction candidates for an encoded block of pixels. Ex-1004, ¶11, ¶15, ¶¶24-25. As Rusert iterates through blocks of pixels in a frame, each block will have its own set of previously-coded motion vectors, specific for that block and different from others. Ex-1004, ¶2 ("pixel blocks"), ¶¶11-12, ¶36, ¶59, Fig. 3g. As blocks of the frame are encoded, the number of blocks previous to the current block increases; therefore, the set of previously-coded motion vectors that were used for previous blocks expands with each encoded block. *See id.*; Ex-1004, ¶59. Blocks were encoded/decoded in sequence, with a common approach being a raster scanning order (from top left to bottom right). *See, e.g.,* Ex-1004, ¶59. (teaching that "the blocks to the right and below the current block position have not been visited yet, and do not yet have motion vectors known for them"), Fig. 3g. I have illustrated this concept below where the current block is indicated in black. Each time the current block advances, the set of spatial motion vector prediction candidates increases by one.



111. Regardless of how Rusert iterates through the blocks of pixels in a frame, the set of spatial motion vector prediction candidates is different for different blocks of pixels. The decoder applies these teachings to an encoded block of pixels. As Rusert explains, the decoder receives encoded blocks from the encoder (Ex-1004, ¶¶34-36, Fig. 1) and then "follow[s] the same rules" as the encoder to decode those blocks, building and using the same list of PMV candidates for each block. Thus, Rusert determines a set of spatial motion vector prediction candidates for the encoded block of pixels.

112. Rusert teaches **the spatial motion vector prediction candidates being provided with motion information**. For example, Rusert teaches that "each PMV candidate corresponds to a motion vector used for coding of a previous block." Ex-1004, ¶11; *see also* Ex-1004, ¶¶24-25, ¶¶39-41. The motion vectors of

the PMV candidate include, for example, "x and y components." Ex-1004, ¶106, ¶36, ¶¶91-94, ¶100. Therefore, Rusert's PMV candidates include motion information, including motion vectors and their x and y components, of the PMV candidates. *See* Ex-1004, ¶¶2-3 (discussing motion vectors used for prediction of pixel blocks across frames); *see also* Ex-1001, 2:48-60 ("motion information is indicated by motion vectors associated with each motion compensated image block").

113. Zheng teaches that candidates in a merge list include motion information by which an "encoder instructs a decoder, through bitstream signaling of prediction syntax, to copy a motion vector, reference index (identifying a reference frame, in a given reference picture list, to which the motion vector points) and the motion prediction direction (which identifies the reference picture list (List 0 or List 1), i.e., in terms of whether the reference frame temporally precedes or follows the currently frame) from a selected candidate motion vector for a current portion of the frame." Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27). Thus, for the combination of Rusert and Zheng the candidates in the list include motion information. For example, the information specifying a motion vector would include the x and y components of the motion vector. Ex-1004, ¶106; Fig. 5. Rusert teaches that candidates include motion information, and those candidates are stored in Rusert's PMV_CANDS, which is a merge list. *Supra*

§V.A.3[9a]. Thus, in the combination of Rusert and Zheng, regardless of whether it is referred to as a PMV_CANDS or merge list, the candidates include motion information.

114. Zheng teaches that "[m]erge candidates are generated per PU. That is, each PU has its own set of merge candidates." Ex-1005, ¶47; Ex-1055, 000018 (¶29). Zheng teaches that at the decoder, the motion information may have already been determined for a causal block and thus "the decoder may simply retrieve the motion vector, reference index, and motion prediction direction associated with the causal PU from memory and copy these values for the current PU [block]." Ex-1005, ¶41; Ex-1055, 000018 (¶27). Therefore, Zheng expressly teaches that the spatial motion vector prediction candidates are provided with motion information. More specifically, Zheng teaches various information related to a prediction unit for motion vector prediction. For example, Zheng teaches that "coding a PU using inter-prediction involves calculating a motion vector between a current block and a block in a reference frame." Ex-1005, ¶38; Ex-1055, 000017 (¶24).

115. Additionally, a POSITA would have understood that, in H.265 adopted by Zheng, a PU may include data defining a motion vector, which includes a horizontal component (e.g., x-component), a vertical component (e.g., y-component), a resolution, a reference frame, and/or a reference list. Ex-1011, ¶35. It would have been obvious to include the PU information mentioned above in

PMV_CANDS because the combination relies on Zheng's PU teachings and applies them to Rusert. *Supra* §V.A.3[9a] (explaining how Rusert and Zheng would have been combined).

116. Based on the foregoing explanations, Ground 1 teaches limitation [9b].

[9c]	**selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

117. Ground 1 teaches limitation [9c], as explained below.

118. Ground 1 teaches **selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates** (e.g., a set of previously-coded motion vectors) **as a potential spatial motion vector prediction candidate to be included in a merge list** (e.g., a "merge candidate set"). For example, Rusert teaches "selecting … PMV candidates" for a current block from a "set of previously coded motion vectors that were used for previous blocks[.]" Ex-1004, ¶11, ¶12, ¶15, ¶¶24-25, ¶39, ¶113, Fig. 6. The PMV candidates are selected in a "scan… around the current block" for potential inclusion in PMV_CANDS. Ex-1004, ¶44, ¶¶51-66. Rusert teaches numerous examples (*see* Ex-1004, Figs. 3a-n), including the "search pattern shown in FIG. 3n" that "has been found to combine good compression efficiency (i.e.,

finding good motion vectors) with good computation performance." Ex-1004, ¶66, Fig. 3n:

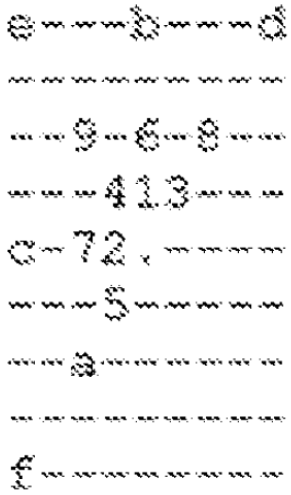


Fig. 3n

119. I note that Fig. 3n illustrates a scan pattern with "an ordering as shown": 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f with "a" being hexadecimal for 10, "b" being hexadecimal for 11, and so on. Ex-1004, ¶65.

120. Rusert selects candidates in the sequences taught by its scan patterns, starting with a first candidate labelled "1" in its figures. Ex-1004, ¶¶51-66, Figs. 3a-n. Each iteration of Rusert's algorithm selects a new spatial motion vector candidate to consider. Claim 9 is directed at a single iteration of the method described. Each candidate is selected from the set of previously-coded motion vectors that were used for previous blocks. Ex-1004, ¶11, ¶12, ¶15, ¶¶24-25, ¶39, ¶44, ¶¶51-66. Rusert teaches visiting previously-coded blocks, in the sequences

shown by its scan patterns, and selecting the motion vector for that previously-coded block as a candidate for potential inclusion in PMV_CANDS. Ex-1004, ¶44, ¶¶51-66; *infra* §§V.A.3[9c]-[9d] (explaining how the selected candidate is compared with candidates in PMV_CANDS for potential inclusion in PMV_CANDS). I have illustrated the start of the scan patterns for Fig. 3n (horizontally):

121. Rusert teaches that the candidate list is for the prediction unit. "The PMV_CANDS list [is] used for coding a motion vector associated with a current motion compensation block[.]" Ex-1004, ¶41; *see also* Ex-1004, ¶4, ¶39, ¶42. PMV_CANDS is "dynamically generated specifically for the current motion compensation block[.]" Ex-1004, ¶41. "In that case, before a block is processed, a PMV_CANDS list is initialized and then updated with a number of previously coded or pre-defined motion vectors." *Id*. When Rusert's PMV_CANDS list is complete, a "predicted motion vector (PMV)" that "is used to predict a [motion vector]… is signaled" using PMV_CANDS and an index "to select a particular PMV candidate… from… PMV_CANDS[.]" Ex-1004, ¶36.

122. Ground 1 applies Zheng's teachings of a merge list to PMV_CANDS. *Supra* §V.A.3[9a] ("Motivation to Combine" subsections explaining how and why Rusert and Zheng's teachings would have been combined).

123. Zheng teaches **a merge list for the prediction unit**. For example, Zheng teaches that "[m]erge candidates are generated per PU. That is, each PU has its own set of merge candidates." Ex-1005, ¶47; Ex-1055, 000018 (¶29). Zheng further teaches that "the encoder may select a set of candidate motion vectors determined for spatially neighboring blocks in the same frame as the current PU or a candidate motion vector determined for a co-located PU in a reference frame." Ex-1005, ¶40; Ex-1055, 000017 (¶25). From this set of merge candidates, "[t]he encoder may perform motion vector prediction to select a particular candidate motion vector to reduce bit rate in signaling." Ex-1005, ¶40; Ex-1055, 000017 (¶25). Zheng teaches that the motion vector predictor is signaled to the decoder through "an index into a candidate motion vector set identifying the selected candidate motion vector" and that is specifically done in merge mode by "include[ing] a flag identifying the mode (in this case 'merge' mode) and an index identifying the selected candidate motion vector." Ex-1005, ¶41; Ex-1055 000017-18 (¶¶25-27).

124. As I explained above, the prediction unit of Rusert's block of pixels is the block itself because Rusert's block is the unit for which a motion vector is

assigned for motion prediction. *See supra* §V.A.3[9a]. For Rusert's teachings, a motion vector is provided for each "8x8 pixel block," also called a "sub-block" because it is a portion of a "macroblock." Ex-1004, ¶36. When Rusert looks for neighboring motion vectors, it scans nearby blocks where motion vectors have been assigned to those blocks. Ex-1004, ¶¶50-67, Figs. 3a-3n. Those blocks are therefore the prediction units in the context of Rusert's teachings. As I also explained above, the combination of Rusert and Zheng applies Rusert's teaching of "blocks," an H.264 concept, to "prediction units," a H.265 concept. *See supra* §V.A.3[9a].

125. **Where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates**. As I explained above, Zheng teaches a merge candidate list that is created using merge candidates. Ex-1005, ¶47; Ex-1055, 000018 (¶29). The merge candidate list comprises merge candidates, which include motion information for the merge candidates, i.e., motion information of the spatial motion vector prediction candidates. Ex-1005, ¶¶38, 41; Ex-1055, 000017-18 (¶¶24, 27). For example, Zheng teaches how to generate the merge candidate set which is specifically based on the motion information of each candidate. Ex-1005, ¶¶51-56; Ex-1055, 000019 (¶35). The selected motion vector is signaled via an index that indicates both the motion vector and reference index. Ex-1005, ¶41; Ex-1055, 000017-18 (¶¶26-27). Zheng

teaches that motion information may contain the motion vector, a reference frame, and prediction direction. Ex-1005, ¶109; Ex-1054, 000006 (¶2); *see also* Ex-1001, 2:65-3:8 (describing motion field information, which includes motion vector and reference picture index). Zheng also teaches that the merge list is based on comparisons of the motion information of the spatial motion vector candidate. Ex-1005, ¶¶51-56 (explaining how merge candidates are compared, and if the motion information is the same, the candidate is not included in the candidate set); Ex-1055, 000019 (¶35). Therefore, Zheng teaches that the merge list is constructed based on the motion information of the spatial motion vector prediction candidates. Further, as I explained above, POSITA would have been motivated to apply Zheng's merge list teachings to Rusert, with a PMV candidate being used as a merge candidate and PMV_CANDS being used as the merge list.

126. Likewise, Rusert teaches that the PMV_CANDS list comprises motion information of the spatial motion vector prediction candidates. Like Zheng's teaching that the merge candidate set includes candidates for predicting the motion of the current block, Rusert teaches that PMV_CANDS is a "list of PMV candidates" for predicting the motion of the current block. Ex-1004, ¶37. Furthermore, Rusert teaches that "each PMV candidate corresponds to a motion vector used for coding of a previous block." Ex-1004, ¶11; *see also* Ex-1004, ¶¶24-25, ¶39, ¶41. The motion vectors of the PMV candidate include, for

example, "x and y components." Ex-1004, ¶106, ¶36, ¶¶91-94, ¶100. Therefore, the PMV_CANDS list includes motion information, including motion vectors and their x and y components, of the PMV candidates. *See* Ex-1004, ¶¶2-3 (discussing motion vectors used for prediction of pixel blocks across frames); *see also* Ex-1001, 2:48-60 ("motion information is indicated by motion vectors associated with each motion compensated image block"). Rusert teaches PMV_CANDS is updated with PMV candidates based on comparisons involving motion information of the PMV candidates. Ex-1004, ¶¶71-72; *infra* §V.A.3[9e]. Therefore, PMV_CANDS is constructed based on the motion information of the PMV candidates.

127. Ground 1 further applies Rusert's block-based teachings to PUs. *Supra* §V.A.3[9a] (explaining how and why the references would be combined).

128. Based on the foregoing explanations, Ground 1 teaches limitation [9c].

[9d]	**determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

129. Ground 1 teaches limitation [9d], as explained below.

130. Rusert teaches **determining a subset of spatial motion vector prediction candidates** (e.g., based on Rusert's scan of previous PMV candidates within a specified distance and allowable relative position). For example, Rusert

teaches "selecting a set of PMV candidates as a *subset* of the set of previously coded motion vectors that were used for previous blocks[.]" Ex-1004, ¶11, ¶12 ("A PMV candidate list is created by *selecting a subset of the motion vectors* previously used for previous blocks"), ¶¶24-25; *supra* §IV.D.

131. Rusert determines the subset of PMV candidates based on the locations of their corresponding blocks: "the selected set of PMV candidates comprises *a subset of the set of previously coded motion vectors* … having an allowed distance from the current block and an allowed position[.]" Ex-1004, ¶15, ¶37, Figs. 2a-2b. For example, following the scan pattern of Fig. 3n with an allowed distance of 1 results in a scan consistent with that for a merge list. Ex-1004, ¶15, ¶37, Fig. 3n:

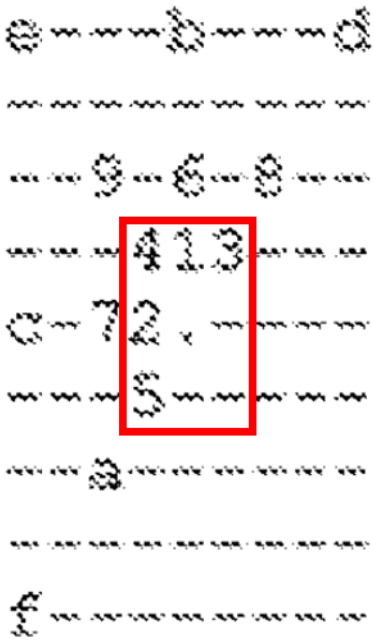


Fig. 3n

132. When Rusert's subset of PMV candidates, which is stored in PMV_CANDS, is updated with a PMV candidate (Ex-1004, ¶39 ("Update means that one or more motion vectors are added to an existing [PMV_CANDS] list")) using a scan pattern for the current block, the subset of PMV candidates comprises the previously-coded PMV candidates obtained from blocks in previous locations of Rusert's scan pattern from the initial block (e.g., 1) up to the candidate being evaluated (e.g., 3). Ex-1004, ¶¶43-44, ¶¶51-66, ¶¶4-5, ¶¶36-39 ("A PMV_CANDS list may be updated to include previously coded motion vectors MV."). As the scan progresses, the subset includes the PMV candidates obtained from blocks in previous locations of Rusert's scan pattern. Ex-1004, ¶44, ¶¶51-66, Figs. 3a-3n.

For example, following the scan pattern of Fig. 3n, when block "3" is selected in limitation [9c], the subset of PMV candidates includes PMV candidates obtained from blocks located at positions 1 and 2, as illustrated below. *Id.* When block "4" is selected, the subset of PMV candidates includes PMV candidates obtained from blocks located at positions 1, 2, and 3, and so on. *Id.*

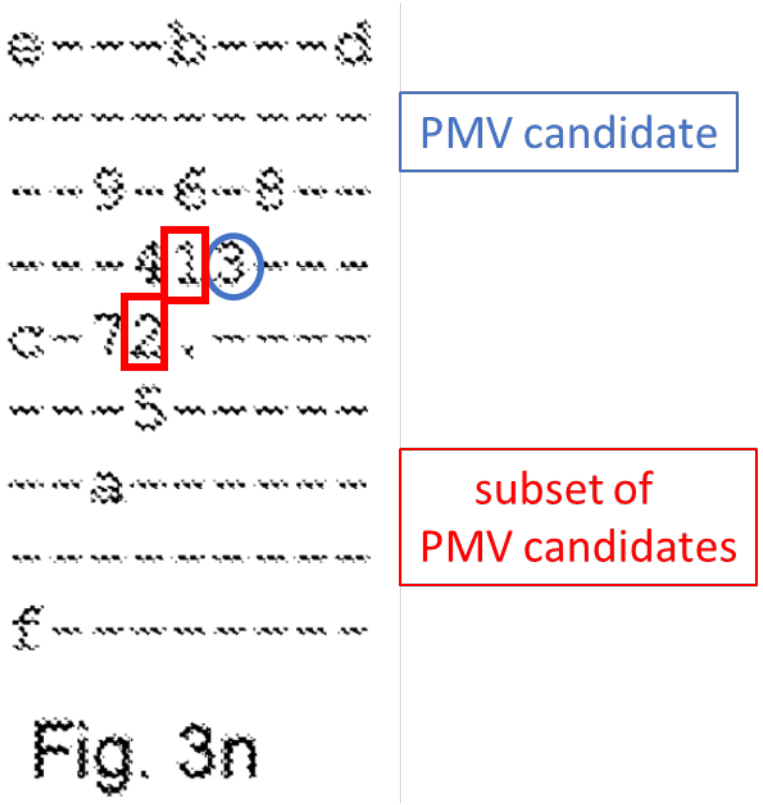


Fig. 3n

133. The PMV candidates in Rusert's subset of PMV candidates are a subset of the larger set of previously-coded motion vectors. Additionally, Rusert teaches terminating a scan "as soon as a pre-defined number of unique PMV candidates have been found[.]" Ex-1004, ¶48. Therefore, not all PMV candidates are considered. Even PMV candidates that are within "a certain distance" (e.g., 1)

and part of "a predetermined scan pattern" are not considered if the "pre-defined number of unique PMV candidates have been found." Ex-1004, ¶¶44-49. For example, while Fig. 3n includes a scan of up to 15 blocks, Rusert terminates the scan with a subset of candidates from those blocks when a pre-determined number of candidates are obtained (e.g., 4), without using the remaining candidates in the scan sequence. Ex-1004, ¶48, ¶107. The subset of PMV candidates is therefore a subset of the set of PMV candidates that are within a certain distance and part of the scan pattern. As the scan progresses and the subset of PMV candidates is updated, the subset is stored as a list of PMV candidates called PMV_CANDS. Ex-1004, ¶¶37-41, ¶¶44-49, ¶¶51-66; *infra* §V.A.3[9d]. The determined subset is then used when performing the comparisons of motion information described in limitations [9e]-[9f], as will be explained below.

134. Rusert improves coding efficiency by using this subset of spatial motion vector prediction candidates instead of all of the previously-coded motion vectors. Ex-1004, ¶13 ("The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use.… This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long."), ¶12 ("*Restricting the previous blocks* that are considered reduces the number of previous motion

vectors that must be considered *meaning that less computation is needed, improving the processor efficiency of the coding*.").

135. The determination is **based on the location of the block associated with the first spatial motion vector prediction candidate**. Here, "the" block recited in [9d] has an antecedent basis in [9a]: "receiving an encoded *block of pixels including a prediction unit*[.]" The same antecedent basis in [9a] applies to limitation [9b], which recites "determining a set of spatial motion vector prediction candidates for *the encoded block of pixels*[.]" Therefore, "the block" in limitation [9d] refers to the "block of pixels" in [9a] and [9b] and therefore refers to the current block for which motion vector prediction candidates are being analyzed.[2] *Supra* §IV.C.

136. Rusert determines its subset based on the location of the current block. The subset of PMV candidates is updated by a "scan… around the current block[.]" Ex-1004, ¶44, ¶43 ("Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by *using motion vectors associated with*

[2] This is further confirmed by the prosecution history. The Examiner interpretated "based on the location of the block associated with the first spatial motion vector prediction candidate" as based on the location of the "current block," finding that this limitation was satisfied by prior art teachings for "selecting one of the PMV from MVs of neighboring blocks[.]" Ex-1002, 000198 (Non-Final Office Action). The Applicant did not dispute this plain reading of the claims. *See* Ex-1002, 000785-788 (Reply to Office Action).

surrounding blocks.”). Because the scan is “performed around the current block[,]” the set of blocks that are scanned for potential inclusion in the subset of PMV candidates is based on the location of the current block within the frame, e.g. block N. Ex-1004, ¶44. The set of scanned blocks is determined relative to the location of the current block within the frame, meaning scanned blocks 1, 2, 3, 4, and 5 refer to *different* blocks within the frame when the current block changes. Therefore, the location of the current block is associated with the candidates.

137. Furthermore, the subset of PMV candidates is based on “an allowed distance from the current block and an allowed position.” Ex-1004, ¶15, ¶11, ¶13, ¶17, ¶¶24-25, ¶113, Fig. 6. The distance and position are relative to and therefore based on the location of the current block. *See, e.g.*, Ex-1004, ¶11 (“The method comprises identifying allowed distance values of distances between the current block and the previous block.”), ¶16 (discussing identifying allowed positions relative to the current block). Rusert teaches several techniques for calculating distance values based on the locations of the current block and surrounding blocks, including Euclidean distance,[3] Manhattan distance, and Chebyshev distance. Ex-1004, ¶44. Within the allowed distance, Rusert teaches scanning surrounding blocks, starting with closest blocks and terminating the scan “once a certain

[3] Rusert occasionally misspells “Euclidean distance” as “Euclidian distance”. *See, e.g.*, Ex-1004, ¶44, ¶78, ¶87.

distance has been reached." This prevents blocks from the set that are outside of the allowed distance from being scanned and included in the subset. Ex-1004, ¶13 (blocks with "certain distance values" outside the allowed distance are not included in the subset), ¶43, ¶47. Rusert's figures include exemplary distance values of surrounding blocks relative to the current block, with the scan terminating when a certain distance value, such as 1, is reached. *E.g.* Ex-1004, Figs. 3d, 3f, 3g:

```
22222
21112
21.12
21112
22222
```

Fig. 3d

```
3333333
3222223
3211123
321.123
3211123
3222223
3333333
```

Fig. 3f

```
3333333
3222223
3211123
321.---
321----
322----
333----
```

Fig. 3g

138. In each of Rusert's exemplary scans, the blocks are identified, ordered, and associated with spatial motion vector prediction candidates based on their location around the current block (indicated by "."). Ex-1004, ¶¶51-66, Figs. 3a-3n. Therefore, the subset of PMV candidates in PMV_CANDS is based on the current block, which is associated with the first spatial motion vector prediction candidate selected in limitation [9c], e.g., the candidate spatial motion vector prediction being evaluated (e.g., block "3"). Ex-1004, ¶11, ¶13, ¶15, ¶17, ¶¶24-25, ¶¶43-44, ¶¶51-66.

139. Additionally, Rusert teaches that only motion vector prediction candidates from blocks with allowed positions relative to the current block will be included in the subset. *See* Ex-1004, ¶15. Rusert explains:

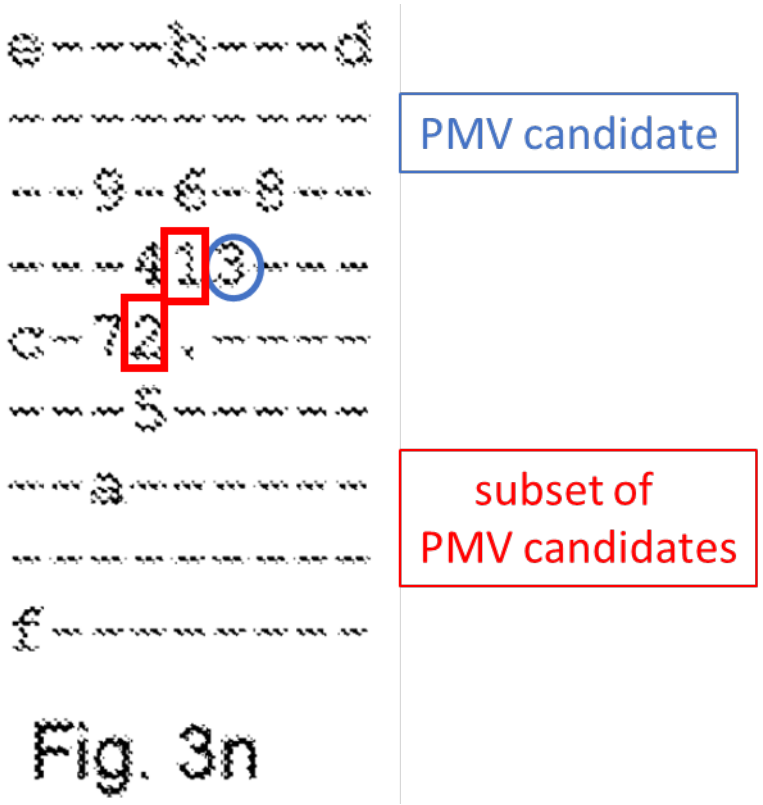
> The allowed positions may be are *corner and middle block positions*. A previous block at a corner position is offset from a current block position by an equal distance in a horizontal direction and a vertical direction. A previous block at a middle position is either horizontally aligned or vertically aligned with a current block position. In a system where coding is performed by horizontal lines of pixels starting in the top left corner, the allowed block positions may comprise: *blocks horizontally aligned with and offset to the left of the current block; blocks vertically aligned with and offset above the current block; and blocks offset to the left and above the current block by the same distance*. The allowed distance values and/or the allowed positions may be predetermined.

Ex-1004, ¶16 (emphasis added). Here, Rusert teaches that PMV candidates of previous blocks that do not have an allowed position are excluded from the subset. For example, Rusert teaches excluding "all blocks other than the corner blocks and the blocks directly above and to the left of the current block[.]" Ex-1004, ¶65. This exclusion removes blocks that have not yet been encoded/decoded because the "blocks to the right and below the current block position have not been visited yet, and do not yet have motion vectors known for them[.]" Ex-1004, ¶59. Rusert teaches that excluding such blocks prevents "requir[ing] too many tests before

growing the scan area big enough to capture distant blocks." Ex-1004, ¶65. Therefore, the subset of PMV candidates in PMV_CANDS is based on the location of the current block within the frame. When the current block that is being encoded changes, the subset will be different because that block has a different location within the frame.

140. Alternatively, if "the block" were interpreted to mean the block of pixels from which the selected first candidate was obtained for a given iteration, Rusert also teaches this. *Supra* §IV.C. For example, Rusert teaches that a "position" of a block of a PMV candidate relative to the current block (".") is represented as (xpos, ypos), denoting a coordinate comprising an x-coordinate and a y-coordinate of the block, i.e., **the location of the block associated with the first spatial motion vector prediction candidate** within the frame. *See, e.g.*, Ex-1004, ¶¶51-52 (explaining that if the current block that is being encoded is at xpos=3 and ypos=5, then a candidate spatial motion vector from the block at xpos=2 and ypos=1, the Euclidean distance would be the square root of 17). Rusert teaches that the subset of PMV candidates with which the selected candidate is compared against includes the PMV candidates of blocks located in the scan pattern previous to the selected candidate, which is based on the location of selected candidate's block. Ex-1004, ¶44. For example, following the scan pattern illustrated in Fig. 3n, the PMV candidate associated with the block located

at “3” would be compared with a subset of PMV candidates that include the PMV candidates of blocks “1” and “2,” which I illustrate below. *Id.,* ¶¶65-66, Fig. 3n:



141. The subset is determined for each block that is encoded because the set of scanned blocks is determined relative to the location of the current block within the frame (represented by the dot in Fig. 3n). When the current block that is being encoded changes, the scanned blocks refer to different blocks within the frame. Different subsets will be determined for blocks of encoded pixels at two different locations within a frame. For example, even though the scan order may select the block located at “3,” the actual location of block “3” within the frame depends on the location of the current encoded block and will thus have a different

location within the frame when the location of the current block that is being encoded changes. In other words, the absolute x-, y- position of blocks 1, 2, 3, 4, and 5 within the frame changes when the block that is being encoded changes, even if the relative position is the same.

142. Based on the foregoing explanations, Ground 1 teaches limitation [9d].

[9e]	**comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the [sub]set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric; (incorporating certificate of correction)**

143. Ground 1 teaches limitation [9e], as explained below.

144. Rusert teaches **comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the [sub]set of spatial motion vector prediction candidates** (e.g., a subset of previously-coded motion vectors within a specified distance and allowable relative position) **… wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric**. For example, when updating the candidates in the merge list, which is stored as a list in

PMV_CANDS, Rusert teaches at least three comparisons for determining whether to include or skip a candidate (Ex-1004, ¶¶71-72), any one of which is sufficient on its own to satisfy this limitation. When considering a candidate for potential inclusion in the merge list, the "PMV candidate may be determined to be unnecessary if it at least one of the following conditions is fulfilled: [1] the PMV candidate is a duplicate of another PMV candidate in the set; [2] the PMV candidate is determined to be within a threshold distance of an existing PMV candidate; and [3] the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits." Ex-1004, ¶21.

145. First, Rusert teaches performing an equivalence check by comparing whether "the PMV candidate is a duplicate of another PMV candidate" in the list. *Id.*, ¶71. "This ca[n] be done, when updating the list, by comparing the candidates already in the list with the new vector that could be added, and if a duplicate is found… [i]t is preferable to skip the new vector[.]" Ex-1004, ¶71. Skipping (thus excluding) duplicates is "[o]ne measure for reducing the number of candidates[.]" *Id.*; *see also* Ex-1004, ¶62 ("It has been identified as advantageous to discard such duplicates"). When a potential candidate is selected and Rusert evaluates whether to update the merge list of candidates, the selected candidate is compared, based on an equivalence check, with the current motion vectors of the candidates that are

already in the merge list of candidates. If it is a duplicate (e.g., equivalent), then the potential candidate is skipped and excluded. For example, two motion vectors are duplicates, or equivalent, when their vector values (e.g., their x values and y values) are the same.

146. Second, Rusert teaches comparing a difference in motion information to a threshold or other similarity metric: Rusert determines whether "the PMV candidate is… within a threshold distance of an existing PMV candidate" in the list. Ex-1004, ¶21. This is accomplished using "a similarity measure" such as "Euclidian distance… or absolute distance." Ex-1004, ¶72, ¶87. For example, Euclidian distance is measured by "$(x_0-x_1)^2+(y_0-y_1)^2$" and absolute distance is measured by "$|x_0-x_1|+|y_0-y_1|$, with (x_0, y_0) and (x_1, y_1) being the pair of motion vectors under consideration." Ex-1004, ¶72. Here, the distance between the pair of motion vectors is a difference in motion information (e.g., x- and y- components of the motion vectors) that is considered for the candidate and each previous candidate that is already in the list. This difference in motion information is compared with a threshold distance to determine whether it is less than a threshold distance.[4] If the "similarity measure [is] smaller than a pre-defined threshold," then the candidate is "[r]emov[ed] or skipp[ed][.]" Ex-1004, ¶72, ¶87.

---

[4] If the two PMV candidates are duplicates, the distance (such as the Euclidean distance and the absolute distance) between the two PMV candidates is zero.

147. Third, Rusert teaches comparing a difference in motion information to a threshold or other similarity metric: Rusert determines whether "at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits." Ex-1004, ¶21. This is accomplished by "removing PMV candidates" that "will never be used" because another candidate in the list facilitates "a bit sequence that is shorter or of the same length compared for all possible motion vectors." Ex-1004, ¶90. In other words, there exists a better candidate that results in more efficient coding, and hence uses less bits to signal to the decoder how to reconstruct the block. Therefore, this comparison involves a similarity metric because candidates are measured by their ability to minimize the size of the bitstream for encoding motion vectors for the current block. The number of bits used to encode a motion vector embodies a difference between the candidate and the motion vector of the current block, and therefore is a metric of the similarity between the candidate vector and the motion vector of the current block.

148. For example, Rusert teaches that "[i]f we want to encode a motion vector, such as MV=(0,2)," then "we can encode… it using…" a first candidate "PMV=(0,2), plus a difference DMV=(0,0)" or a second candidate "PMV=(-1,2), plus a difference DMV=(1,0)[.]" Ex-1004, ¶¶91-94. Rusert teaches that the same process is followed when decoding an encoded block of pixels; therefore, the same set of candidates is constructed for decoding. Ex-1004, ¶35 ("the encoder and

decoder follow the same rules for creating and modifying the set of PMV candidates"). In this example, the first candidate uses a "total number of… 6 bits" to code the motion vector because of its index costs (4 bits) and the difference cost to code the difference between the motion vector and the first candidate (1 bit for the x component difference and 1 bit for the y component difference). Ex-1004, ¶93. The second candidate only uses "5 bits in total" because of its index costs (1 bit) and the difference cost (3 bits for the x- component difference and 1 bit for the y- component difference). From this example, Rusert teaches that "it will never be beneficial to use" the first candidate because the second candidate "will always be one [b]it cheaper or better." Ex-1004, ¶95. Therefore, when evaluating whether the subset of candidates should be updated with a potential candidate, Rusert teaches comparing the motion information of the potential candidate with that of each candidate that is already in the subset of candidates, to determine if one will allow motion vectors to be decoded using fewer bits.

149. All three teachings compare a potential candidate with at least one other candidate from the subset of preceding candidates in Rusert's scan sequence. Since preceding candidates have smaller index values, they would be signaled more efficiently than later duplicates, and thus Rusert improves efficiency by determining this subset and comparing potential new candidates to the subset. Ex-1004, ¶¶88-98. For all three comparisons, the x- and y- components of

Rusert's candidates are compared; this is motion information of the candidates because the x- and y- components of the vector values describe the "motion of pixel blocks across frames." Ex-1004, ¶2. For example, "coding of a motion vector consists of… a motion vector predictor… and … the difference… between the motion vector and the motion vector predictor." Ex-1004, ¶3, ¶¶36-37; *supra* §V.A.3[9a] (explaining that the motion information is represented by vector values). Therefore, each of Rusert's three comparisons are comparing the motion information of a candidate with the motion information of another candidate in the subset of previously-coded motion vectors—in particular the candidates currently in the list of candidates.

150. Additionally, for all three comparisons, Rusert teaches or at least suggests performing these comparisons when evaluating whether to update PMV_CANDS, which stores the list of candidates, with a candidate. Ex-1004, ¶¶71, ¶75, ¶¶84-87, ¶90. Furthermore, this would have been obvious because Rusert teaches the advantages of reducing the number of candidates in PMV_CANDS, including with the use of these comparisons (Ex-1004, ¶12, ¶21, ¶70, ¶84, ¶90), and the natural time to perform these comparisons would have been when evaluating whether or not to add a candidate to PMV_CANDS. As Rusert teaches, performing this check when PMV_CANDS is updated will prevent "unnecessary" candidates from being added, "because it may happen that some

candidates… will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors." Ex-1004, ¶90. This process is also used when decoding an encoded block of pixels; therefore, the same set of candidates is constructed for encoding and decoding. Ex-1004, ¶35. Rusert teaches performing the same process of comparing motion information to form the merge list when decoding an encoded block of pixels. Ex-1004, ¶24, ¶35.

151. Rusert teaches **comparing … without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates**. For example, Rusert teaches that "[r]estricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding." Ex-1004, ¶12. Each of the three comparisons explained above compares motion information from a selected candidate with the candidates in the subset of candidates, without making a comparison of each possible pair from the set of motion vectors used for all previous blocks (i.e., the set of all spatial motion vector prediction candidates from previously-encoded blocks). Ex-1004, ¶¶11-13, ¶¶15-16, ¶113. First, Rusert teaches comparing whether "the PMV candidate is a duplicate of another PMV candidate" (Ex-1004, ¶21) in the list of candidates "by comparing the candidates already in the list with the new vector

that could be added" (Ex-1004, ¶71). Therefore, comparisons are made between pairs formed by the potential candidate and the candidates currently in the list of candidates and comparisons are not made of the potential candidate with anything not in the subset of candidates, such as motion vectors of previous blocks that have not been scanned yet or that were already excluded from the subset of candidates because they were not within the required distance or allowable relative position. *Id*. Second, Rusert teaches comparing whether "the PMV candidate is… within a threshold distance of an existing PMV candidate[]" in the subset of PMV candidates. Ex-1004, ¶21. Again, the comparisons made are for pairs formed by the potential candidate and the candidates in the list of candidates and not each motion vector for all previous blocks. *Id*. Third, Rusert teaches comparing whether "at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits." Ex-1004, ¶21. This involves comparisons of a potential candidate with candidates already in the list of candidates and not each motion vector for all previous blocks. *Id*. Even within the scan order, Rusert compares potential candidates with those already in the list of candidates, meaning Rusert avoids comparing each possible pair of candidates from the scan order. *See* Ex-1004, ¶¶11-13. Therefore, comparing motion information of a potential candidate with motion information of the candidates already in the list of candidates, uses less computation than comparing motion information of the

candidate with motion information of all previously decoded blocks and thus improves processor efficiency. Ex-1004, ¶12.

152. Additionally, when motion information of a candidate is compared with motion information of the candidates in the list of candidates, the motion information of the candidate is not compared with each pair of candidates from the set of all previously-encoded blocks because the subset of candidates is limited to candidates "having an allowed distance from the current block and an allowed position." Ex-1004, ¶15; *see supra* §V.A.3[9b]. The candidates of the previous blocks that are not within the allowed distance and not in an allowed position are not in the subset of candidates and, therefore, are not compared.

153. Furthermore, Rusert teaches various scan patterns within the allowed distance, including scan patterns that exclude certain blocks within the allowed distance. *See*, *e.g.*, Ex-1004, ¶64 ("we could use only layers that are powers of two: 1, 2, 4, 8, …."), ¶65 ("all blocks other than the corner blocks and the blocks directly above and the left of the current block are excluded."), Figs. 3m, 3n. Because the allowed distance, the allowed position, and the scan pattern exclude candidates of blocks from consideration for potential inclusion in the list of candidates, a candidate is not compared with these blocks, and there is no comparison of each possible pair of motion vectors from the set of all previously-coded blocks.

154. Moreover, even within the various scan patterns that exclude certain blocks, Rusert teaches that a "scan may be terminated… as soon as a pre-defined number of unique PMV candidates have been found[.]" Ex-1004, ¶¶44-48. For example, Rusert teaches "us[ing] a maximum of four candidates in the [PMV_CANDS] list" or "us[ing] seven" as the maximum. Ex-1004, ¶107. Limiting the maximum number of unique candidates reduces "the number of bits needed to specify the candidate vector" and avoids "the problem that many vectors can be represented using several candidates, which is unnecessary." Ex-1004, ¶107. Therefore, Rusert does not compare each pair of candidates from the set of previously-encoded blocks, or even each pair of blocks within the scan order, because the scan of the previously-encoded blocks is terminated once a pre-defined number of unique candidates are found. Rusert thus teaches claim 9 with the recited "set" being either (a) all of the previously-coded motion vectors relative to the current block, or (b) the full set of candidates from Rusert's scan order, because in both cases Rusert compares the selected candidate with a subset of candidates, without comparing each pair from the set of previously-coded motion vectors or the entire scan order.

155. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, with Rusert's PMV candidates being merge candidates, as taught by Zheng, and Rusert's

PMV_CANDS list being a merge list. *Supra* §§V.A.3[9a], [9c]. It would have been obvious to a POSITA to use Zheng's merge mode when using Rusert's algorithm to identify candidate motion vectors and to use Rusert's PMV_CANDS list as a merge list.

156. Indeed, Zheng provides express motivation to exclude redundant candidates in the construction of a merge list. Zheng teaches that "[c]urrent proposals for the HEVC standard implement techniques to avoid using redundant merge candidates for PUs of the same CU." Ex-1005, ¶49; Ex-1055, 000019 (¶33). A POSITA would have understood that a "redundant" candidate is an unnecessary candidate, for example, because it is duplicate or within a threshold distance with another candidate. Zheng teaches this explicitly: "A redundant merge candidate is a merge candidate that has the same motion information as another PU in the same CU." Ex-1005, ¶49; Ex-1055, 000019 (¶33). Therefore, Zheng also teaches comparing motion information of the first merge candidate with motion information of another merge candidate of the merge list, providing a further motivation for this limitation to combine with Rusert.

157. Based on the foregoing explanations, Ground 1 teaches limitation [9e].

[9f]	**if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and**

158. Ground 1 teaches limitation [9f], as explained below.

159. The combination of Rusert and Zheng teaches **if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list**. As I explained above, Rusert teaches at least three ways for comparing motion information of a candidate with motion information of previous candidates that are already in the list, which stores the subset of candidates. *Supra* §V.A.3[9e]. Rusert performs all three comparisons to determine whether a candidate corresponds with (e.g., is equivalent to, is within a threshold difference of, would never be used because of) another candidate in PMV_CANDS and, therefore, whether to include or exclude the selected candidate in PMV_CANDS. Ex-1004, ¶¶20-21.

160. Rusert teaches removing "unnecessary PMV candidates from the set of PMV candidates." Ex-1004, ¶20. "This ensures the length of the [PMV_CANDS] list is not unnecessarily long, which would reduce coding efficiency." Ex-1004, ¶21. "*A PMV candidate may be determined to be*

unnecessary if it at least one of the following conditions is fulfilled: [1] the PMV candidate is a duplicate of another PMV candidate in the set; [2] the PMV candidate is determined to be within a threshold distance of an existing PMV candidate; and [3] the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits." Ex-1004, ¶21. "Unnecessary PMV candidates are removed … because it may happen that some candidates in the list will never be used." Ex-1004, ¶90. In other words, a PMV candidates is unnecessary if the PMV candidate corresponds with another PMV candidate already in PMV_CANDS because the PMV candidate would never be used in coding the motion vector for the current block. Ex-1004, ¶21, ¶90. A shorter PMV_CANDS list "allows the remaining PMV candidates to be signaled using shorter codes and so fewer bits[.]" Ex-1004, ¶¶22, 90 ("thereby making the list shorter and the average bit length of each index shorter.").

161. Rusert excludes unnecessary candidates, including the three categories explained above (duplicate candidates, candidates within a threshold distance of candidates in the PMV_CANDS list, and candidates that have at least one alternative candidate that will allow motion vectors to be coded using fewer bits), when deciding whether to add a new candidate to PMV_CANDS. Ex-1004, ¶21, ¶¶71-72; s*upra* §V.A.3[9e]. As Rusert explains, the "PMV_CANDS list may be

updated to include" candidates. Ex-1004, ¶39, ¶44, ¶71, ¶87; *supra* §V.A.3[9e]. Here, "[u]pdate means that one or more motion vectors are added to an existing PMV_[C]ANDS list." Ex-1004, ¶39. For example, PMV candidates with "unique vectors" are inserted in the PMV_CANDS list based on an "outwards going scan[.]" Ex-1004, ¶50. As part of the "update" process whereby new candidates are added to PMV_CANDS, Rusert determines whether a selected candidate should be included or skipped, meaning it is excluded because it is a duplicate, it is within a threshold distance of another candidate in PMV_CANDS, or there is an alternative candidate that will allow motion vectors to be coded using fewer bits. Ex-1004, ¶¶71-72; *supra* §§V.A.3[9d]-[9e]. Additionally, this would have been obvious to a POSITA because Rusert teaches the advantages of reducing the number of candidates in PMV_CANDS using the three comparisons from [9e] (Ex-1004, ¶12, ¶21, ¶70, ¶84, ¶90), and the natural time to perform the comparisons would have been when evaluating whether or not to add a candidate to the list of candidates stored in PMV_CANDS. As Rusert teaches, performing this check when PMV_CANDS is updated prevents "unnecessary" candidates from being added, "because it may happen that some candidates… will never be used[.]" Ex-1004, ¶90. Thus, Rusert teaches excluding a candidate from PMV_CANDS if at least one of the three comparisons indicates the candidate's motion vector information corresponds with that of another candidate in the list of candidates.

162. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teaching above with respect to PMV_CANDS applies to a merge list, in view of Zheng's teachings of the merge mode. For example, the combination of Rusert and Zheng teaches that, if at least one of the comparisons indicates that the motion vector information of the merge candidate candidates correspond with each other, the first merge candidate is excluded from the merge list, which teaches limitation [9f].

163. Indeed, as I explained above, Zheng provides express motivation to exclude redundant candidates in the construction of a merge list. A POSITA would have understood that a candidate with motion vector information that corresponds to another merge candidate is a redundant merge candidate. Thus, Zheng also teaches, if the motion vector information of the merge candidate candidates is determined to correspond with each other, the first merge candidate is excluded from the merge list. Zheng explicitly teaches not including a candidate motion vector if its motion information is the same as a that of a previously-coded PU. Ex-1005; ¶¶51-56; Ex-1055, 000019 (¶35). This is similar to how Rusert

prevents the redundant candidates from being added to the merge list, which provides a further motivation to combine with Rusert for teaching this limitation.

164. Based on the foregoing explanations, Ground 1 teaches limitation [9f].

[9g]	**selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

165. Ground 1 teaches limitation [9g], as explained below.

166. Rusert teaches **selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels**. For example, Rusert teaches "a code 'index'… to select a particular PMV candidate… from a list of PMV candidates, PMV_CANDS[.]" Ex-1004, ¶36. Then, "[u]sing the transmitted index, the decoder… can determine the PMV… and thus may reconstruction [the motion vector.]" Ex-1004, ¶37. These techniques are "for video decoding, wherein the current block is the block being… decoded[.]" Ex-1004, ¶23, ¶35("The methods disclosed herein are performed… in the decoder during decoding."). The current block is part of an "encoded video stream" that is "passe[d] to a decoder… employed in decoding the encoded video stream." Ex-1004, ¶34. Therefore, Rusert teaches a decoder using an index to select a candidate from PMV_CANDS for use in decoding an encoded block.

167. Rusert teaches **wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate**. As I explained above, Rusert teaches receiving a "a code 'index'" sent "from the encoder… to the decoder" that identifies "a particular PMV candidate … from a list of PMV candidates, PMV_CANDS[.]" Ex-1004, ¶36, ¶37, Fig. 1; *Supra* §V.A.3[9a]. The index is information used by Rusert's decoder to "select a particular PMV candidate… from a list of PMV candidates, PMV_CANDS[.]" Ex-1004, ¶¶36-37. "Using the transmitted index, the decoder… can determine the PMV 220" to "reconstruct [the motion vector.]" Ex-1004, ¶27.

168. Rusert illustrates this process in Figs. 2a and 2b, reproduced below, where Fig. 2a shows the encoding process and Fig. 2b shows the decoding process:

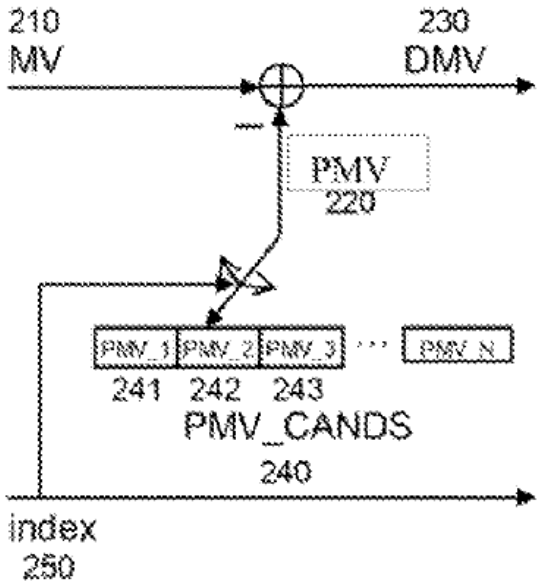


Fig. 2a

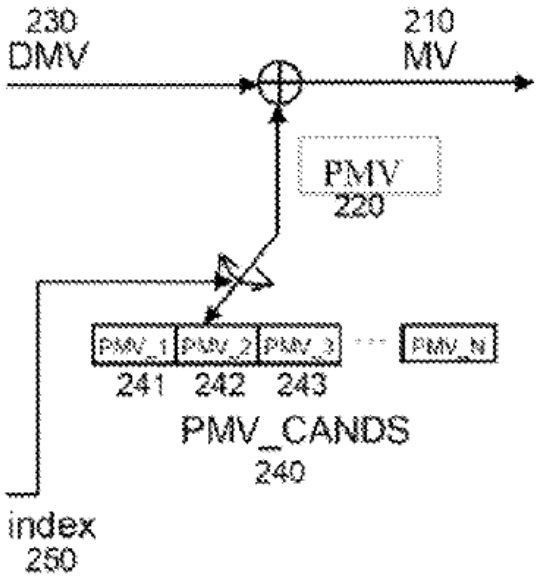


Fig. 2b

This example shows how "a code 'index' 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240[.]" Ex-1004, ¶36. Here, "PMV_CANDS 240 is identically available at both the encoder… and the decoder[]" and "[u]sing the transmitted index, the decoder… can determine the PMV 220" to "reconstruct [the motion vector.]" Ex-1004, ¶37.

169. **Combining Rusert and Zheng**. As I explained above, for example with respect to limitations [9a] and [9c], a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. Accordingly, a POSITA would have understood that Rusert's teaching with respect to the PMV_CANDS list as explained above, in view of Zheng's teachings of the merge mode, applies to a merge list. The combination of Rusert and Zheng teaches selecting a merge candidate from the merge list for use in decoding the encoded block of pixels, wherein the merge candidate is selected from the merge list using the information that was received identifying a respective merge candidate, which teaches limitation [9g].

170. Based on the foregoing explanations, Ground 1 teaches limitation [9g].

4. Dependent Claim 10

10.	**The method according to claim 9 comprising comparing motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.**

171. Ground 1 teaches claim 10, as explained below.

172. As I have explained above, the combination of Rusert and Zheng teaches claim 9 *supra* §V.A.3, including [9e]: comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the [sub]set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates. *Supra* §V.A.3[9e]. Rusert teaches at least three comparisons: (1) whether "the PMV candidate is a duplicate of another PMV candidate" in the PMV_CANDS list; (2) whether "the PMV candidate is … within a threshold distance of an existing PMV candidate"; and (3) whether "at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits." *Supra* §V.A.3[9e]; e.g., Ex-1004, ¶21.

173. Rusert further teaches comparing motion information of the candidate with **at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates**. For example, Rusert teaches a scan

pattern for evaluating whether to include its spatial motion vector prediction candidate in PMV_CANDS. *Supra* §V.A.3[9c]; Ex-1004, ¶44, ¶¶51-66, Figs. 3a-3n. Rusert teaches that the "PMV_CANDS list may be initialized e.g. as an empty list (zero entries)[.]" Ex-1004, ¶39. PMV_CANDS is then "updated to include previously coded motion vectors[.]" *Id*. This update is based on "an outwards going scan… around the current block to obtain motion vectors to update PMV_[C]ANDS." Ex-1004, ¶44. Rusert performs its comparisons, e.g., checking for duplicates, "when updating the list[.]" Ex-1004, ¶71; *supra* §V.A.3[9e].

174. Thus, Rusert begins the scan with a first candidate, e.g., labelled "1" in Figs. 3a-3n, and considers whether to update PMV_CANDS with the first candidate. Ex-1004, ¶39. The scan then moves to the second candidate, which is "compar[ed with] the candidates already in the list[.]" Ex-1004, ¶71. Here, for the second candidate, the PMV_CANDS list includes at most one other spatial motion vector prediction candidate: the first candidate. *See id.* Therefore, the motion information of the second PMV candidate in the sequence is compared with the motion information of at most one other candidate (the first candidate). Ex-1004, ¶21, ¶¶38-40, ¶44. Furthermore, Rusert teaches "the number of candidates in PMV_CANDS may be limited to a pre-defined or dynamically obtained number." Ex-1004, ¶73, ¶¶84-90. With a maximum number of two candidates, the scan would end once two candidates are added to PMV_CANDS, so any potential

candidate would be compared with at most one other candidate, the candidate already in PMV_CANDS. It would have been obvious to a POSITA to choose two as a maximum number because it would allow the PMV candidate to use as motion information for the current block to be signaled with a single bit (e.g., "0" or "1"). *See* Ex-1004, ¶77.

175. Based on the foregoing explanations, Ground 1 teaches claim 10.

5. Dependent Claim 11

11.	**The method according to claim 9 comprising examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.**

176. Ground 1 teaches claim 11 as explained below.

177. As I have explained above, the combination of Rusert and Zheng teaches claim 9. Rusert teaches receiving an encoded block of pixels (the current block). Ex-1004, ¶34, ¶36, Fig. 1; *supra* §§V.A.3[9a], [9e]. Zheng teaches terminology such as prediction units in the H.265 context. *Supra* §V.A.3[9a]. Zheng further teaches excluding a potential spatial motion vector prediction candidate from the merge list when the received encoded block of pixels is divided into two prediction units.

178. Zheng teaches **examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit.** For

example, Zheng teaches that, in H.265/HEVC, "a coding unit may have variable sizes depending on video content. … In particular, each coding unit may be further partitioned into prediction units (PUs) and transform units (TUs)." Ex-1005, ¶29; Ex-1055, 000016 (¶20). Zheng further teaches that a CU that can be partitioned into two horizontally- or vertically- oriented PUs. Ex-1005, ¶47 ("A 2NxN partition is made by dividing a square CU into two horizontally oriented PUs, where PU 0 is on top of PU 1. An Nx2N partition is made by dividing a square CU into two vertically oriented PUs, where PU 0 is to the left of PU 1."), Fig. 2; Ex-1055, 000018 (¶¶29-31), 000003 (Fig. 2), 000004 (Fig. 3A & 3B). When a CU is not divided, the CU corresponds to a single PU. When a CU is divided horizontally, the 2Nx2N square is divided in half into two symmetric PU rectangles having dimensions of 2NxN. Likewise, when a CU is divided vertically, the 2Nx2N square is divided into two symmetric PU rectangles of Nx2N size. Ex-1005, ¶47, Figs. 2, 3A, 3B; Ex-1055, 000018 (¶¶29-31), 000003 (Fig. 2), 000004 (Figs. 3A & 3B). Zheng teaches both examples, where the block of pixels is divided into two prediction units: a first PU ("PU0") and a second PU ("PU1"). Ex-1005, ¶47, Figs. 2, 3A, 3B;[5] Ex-1055, 000018 (¶¶29-31), 000003 (Fig. 2),

[5] Figs. 3A and 3B of Zheng were published by the PTO without exact reproductions of the figures as filed. Ex-1005, Figs. 3A-3B; Ex-1055, 000004 (Figs. 3A-3B). The reproduced figures are from the as-issued version of Zheng,

000004 (Figs. 3A & 3B). This limitation is also satisfied the other way around, where the claimed "first" PU is PU1 and the claimed "second" PU is PU0.

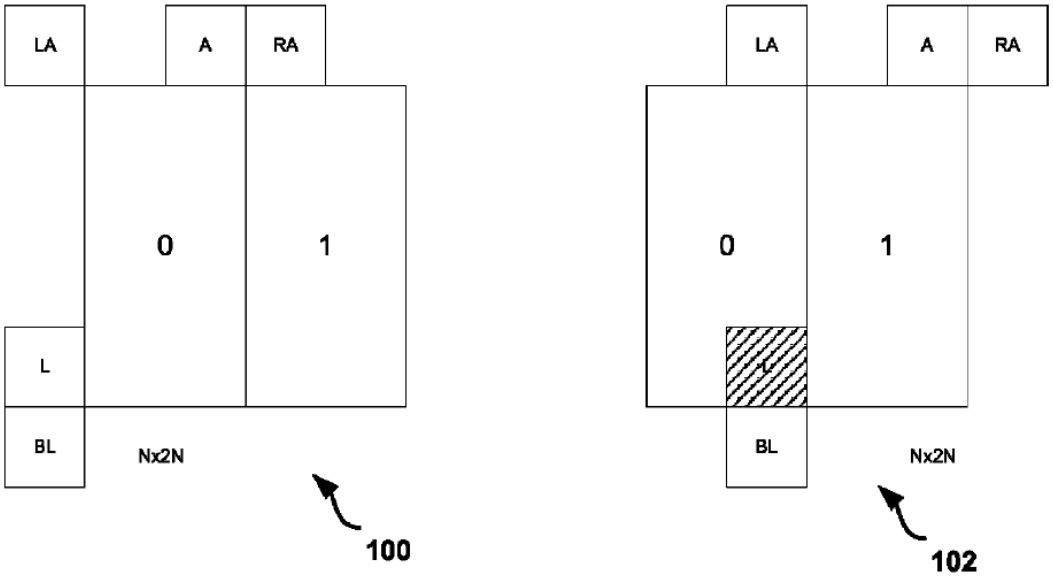


FIG. 3A

U.S. Patent No. 9,313,494. Ex-1061, Figs. 3A-B; *see also* Ex-1062, 000049 (Figs. 3A-B).

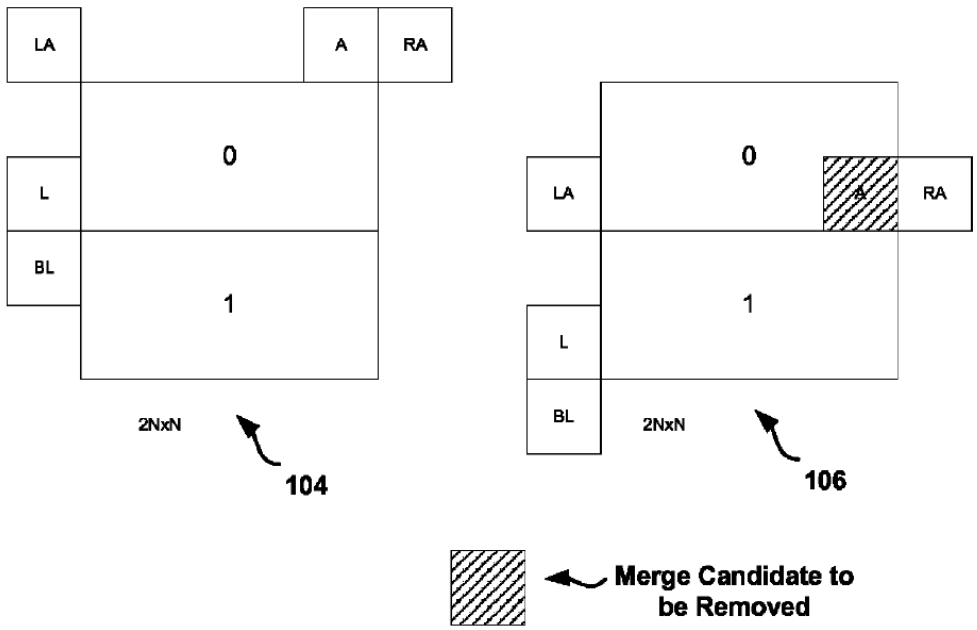


FIG. 3B

179. Zheng teaches that for certain PUs of a partitioned CU that "merge candidates within the merge candidate set for a current prediction unit may be removed/excluded if they are within another prediction unit of the *same coding unit*." Ex-1005, ¶78; Ex-1055, 000018 (¶30) ("As can be seen for merge set 102 for prediction unit 1 of the Nx2N partition, the merge candidate A is from the previous prediction unit 0 of the same coding unit."), 000018 (¶31) ("As can be seen for merge set 106 for prediction unit 1 of the 2NxN partition, the merge candidate B is from the previous prediction unit 0 of the same coding unit."). These scenarios include a CU divided horizontally into two prediction units ("2NxN") or vertically into two prediction units ("Nx2N") where "techniques to avoid using redundant merge candidates for PUs of the same CU. A redundant

merge candidate is a merge candidate that has the same motion information as another PU in the same CU." Ex-1005, ¶49; Ex-1055, 000019 (¶33); *see also* Ex 1005, ¶58; Ex-1055, 000019 (¶33), 000020 (¶36), 000022 (¶44); Ex-1056, 000018 (¶28). For the 2NxN and Nx2N partition types, Zheng teaches that "merge candidates within the merge candidate set for a current prediction unit may be removed/excluded if they are within another prediction unit of the same coding unit." Ex-1005, ¶78; Ex-1055, 000022 (¶44). These teachings apply to an encoded block of pixels received by a decoder because Rusert and Zheng teach the decoder "mimics the encoder in order to achieve encoder/decoder synchronization. Ex-1004, ¶35, ¶¶24-35; Ex-1005, ¶99 ("… the video decoder 30 may implement any or all of these techniques for generating candidate motion vectors in a merge mode in a video decoding process."); Ex-1055, 000026 (¶59) (same). Zheng teaches examining whether a block of pixels is divided horizontally or vertically into two prediction units to exclude redundant motion vector predictor candidates, as further explained below.

180. Zheng teaches **and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.** For example, Zheng examines whether (i) the current block is divided into two PUs and (ii) the spatial motion vector prediction candidate is from the other PU of the same CU; if so, Zheng teaches excluding the

candidate from the merge list. As illustrated in Figs. 3A and 3B below,[6] "merge candidates within the merge candidate set for a current prediction unit may be removed/excluded if they are within another prediction unit of the same coding unit." Ex-1005, ¶78; Ex-1055, 000022 (¶44). Specifically Zheng teaches that for the "2NxN/Nx2N cases: if a merge candidate for the second PU (PU 1) has the same motion information as that of the first PU (PU 0), that merge candidate will be set as not available." Ex-1005, ¶58; Ex-1056, 000018 (¶28).

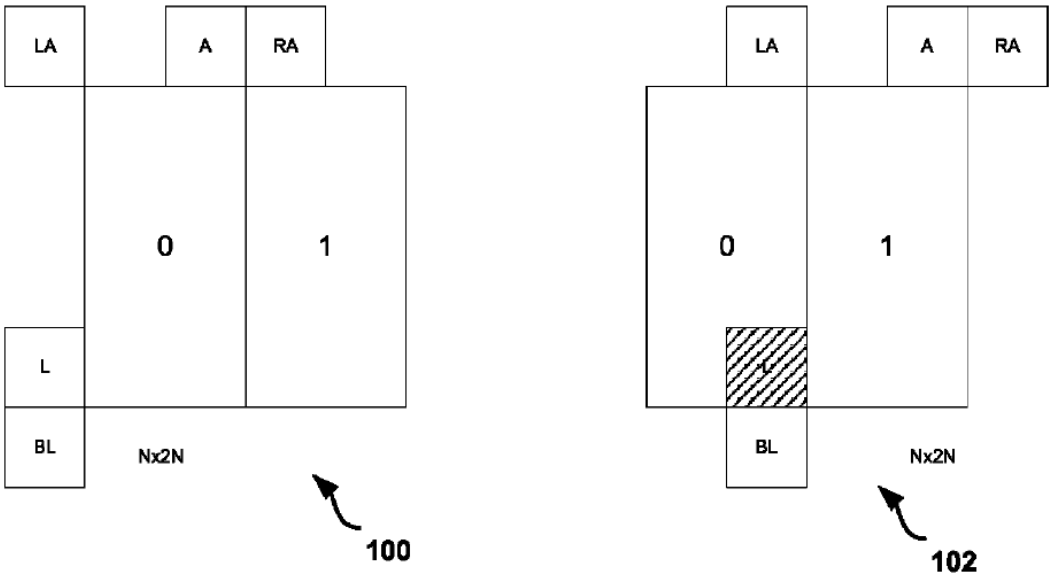


FIG. 3A

[6] *See supra* n.5.

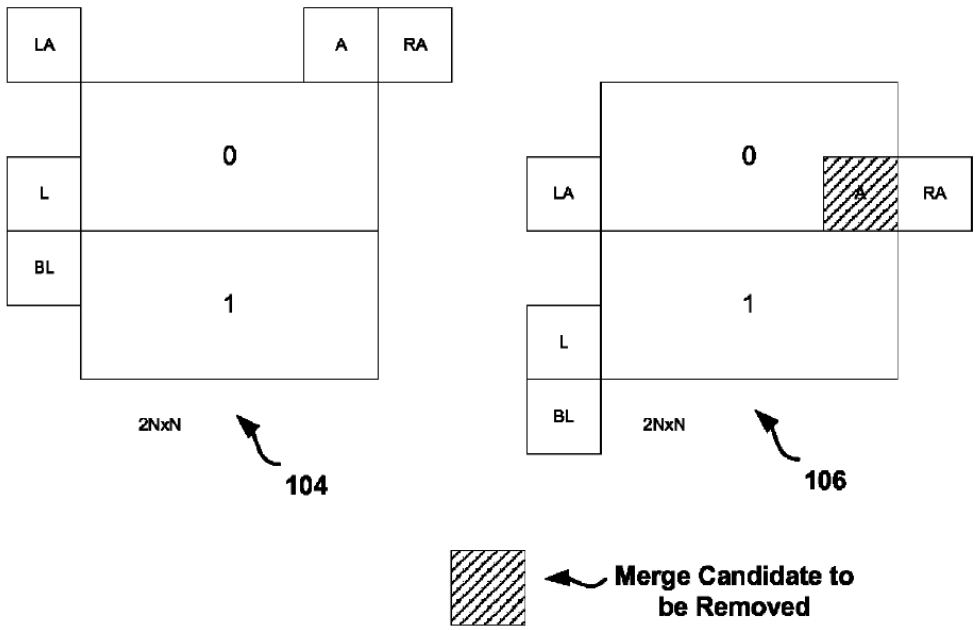


FIG. 3B

181. In particular, Zheng teaches in reference to Fig. 3A, that "for merge set 102 for PU 1, merge candidate L is from PU 0 of the same CU. As such, merge candidate L may be removed/excluded from merge candidate set 102." Ex-1005, ¶70; Ex-1055, 000018 (¶30). And Zheng teaches in reference to Fig. 3B, that "for merge set 106 for PU 1, merge candidate A is from PU 0 of the same CU. As such, merge candidate A may be removed/excluded from merge candidate set 106." Ex-1005, ¶72; Ex-1055, 000018 (¶31). Therefore, Zheng teaches examining whether the block is divided into two PUs (e.g., 2NxN and Nx2N), and if so, removing the MVP candidate of the second PU from the MVP candidate set because it is redundant.

182. This concept was obvious in view of the reason why a CU is divided to begin with. CUs do not have to be divided—in the simplest case, one CU comprises one PU (they are commensurate in size). Ex-1005, ¶29 ("each coding unit may be further partitioned into prediction units (PUs) and transform units (TUs)."); Ex-1055, 000016 (¶20). If the block has uniform motion, there is no reason to divide it into multiple PUs. Conversely, if different parts of the block are moving in different directions, the CU can be divided into multiple PUs so that each can have a different motion vector. Ex-1005, ¶47 ("Merge candidates are generated per PU. That is, each PU has its own set of merge candidates. This includes PUs that are partitioned from a larger CU."); Ex-1055, 000018 (¶29). In other words, the block is divided into two PUs when each half of the block has different motion. If a block is divided into two PUs because its two portions have differing motion vectors, there is no way that the motion vector corresponding to one of these PUs can be a good predictor for the motion of the other PU. Therefore, when analyzing potential motion vectors for half of a block (that has been created by dividing the larger CU into two PUs), Zheng explains that the motion vector from the other half can be removed. Ex-1005, ¶49 ("If every PU in a CU used the same motion information, the result would be duplicative of the 2Nx2N partition type (i.e., no partitioning). Thus the advantage of partitioning a CU for more accurate inter prediction would be negated."); Ex-1055, 000019

(¶33). Zheng explains that assigning the same motion vector to both halves of a divided block is redundant to the scenario where the block was not divided at all and the same motion vector was applied to the entire block comprising one PU. That is why Zheng teaches "avoid[ing] using redundant merge candidates for PUs of the same CU. A redundant merge candidate is a merge candidate that has the same motion information as another PU in the same CU." Ex-1005, ¶49; Ex-1055, 000019 (¶33).

183. It would have been obvious to a POSITA to combine Zheng's teachings with Rusert to remove redundant candidates from PMV_CANDS. *Supra* §V.A.3 (explaining how and why the teachings of the references would have been combined). In particular, a POSITA would have been motivated to apply Zheng's teachings of excluding candidates from the other PU when the received encoded block is divided into two PUs, as explained above, to Rusert's process for selecting candidates. *See id.*

184. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Zheng's teachings of the merge mode,

apply to a merge list. Zheng's teachings are based on the H.265 context, which includes the concept of prediction units. *See supra* §V.A.3[9a]. Zheng provides further motivation to exclude prediction units from merge list because it teaches excluding redundant merge candidates in the construction of a merge candidate list. *See supra* §V.A.3[9d]. Thus, the combination of Rusert and Zheng teaches examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.

185. Zheng further teaches a way to reduce the number of motion vectors that must be considered—particularly for the scenario where a block has been divided into two PUs. Ex-1005, ¶¶49-50, ¶63, ¶78; Ex-1055, 000019 (¶¶33-34), 000021 (¶38), 000022 (¶44). Applying these teachings would have furthered a stated goal of Rusert and Zheng, by reducing the number of motion vector candidates to improve processing and coding efficiency. Rusert seeks to "reduce[] the number of previous motion vectors that must be considered" so "that less computation is needed, improving the processor efficiency of the coding." Ex-1004, ¶12; *see also* Ex-1004, ¶7, ¶13, ¶21, ¶70, ¶84. Zheng likewise seeks to reduce the number of motion vector candidates. Ex-1005, ¶50, ¶¶62-63; Ex-1055, 000019 (¶34), 000020-21 (¶¶37-38).

186. A POSITA would have been motivated to combine Zheng with Rusert because the combination would have used Zheng's known technique of reducing candidates to improve similar devices/methods in the same way. The combination has already applied Zheng's PU teachings to Rusert. Adding Zheng's further optimization of reducing candidates for this PU splitting scenario would have improved similar H.265 PU-based methods. A POSITA would have applied Zheng's known technique to the known devices/methods as taught by Rusert, which was ready for improvement to reduce the number of previous motion vectors that must be considered. Ex-1004, ¶12. Moreover, this would have been a combination of prior art elements according to known methods to yield predictable results.

187. Furthermore, a POSITA would have been motivated by the similarity of the references. While Zheng provides teachings in the context of merge mode, its teachings—e.g., for excluding redundant candidates when a block has been divided into multiple PUs—likewise apply to the combination of Rusert and Zheng, which seeks to efficiently select PMV candidates for a given block (as taught by Rusert) applied to PUs that can come from divided blocks (as taught by Zheng). The same underlying reasoning therefore applies because, even without merge mode, blocks are divided into multiple PUs to assign them different motion vectors, and it would not make sense to divide a block into separate PUs only to

assign the same motion vector to both PUs. Ex-1005, ¶49; Ex-1055, 000019 (¶33). Doing so would be less efficient than keeping the entire block as a single PU and be contrary to the reason why the block was divided to begin with.

188. The combination would have had predictable results: when a block is divided into two PUs, the candidate from one PU is excluded as a candidate for the other PU. Excluding the candidate would have been a predictable result consistent with the reason why a block is divided into PUs to begin with, as explained above.

189. Furthermore, the combination would not have changed the principle of operation for any of the teachings of the references. The combination applies teachings in the manner taught by each reference: Rusert was explained above, and Zheng's teachings are used to exclude candidates when a block is partitioned into two PUs. Ex-1005, ¶¶49-50, ¶63, ¶78; Ex-1055, 000019 (¶¶33-34), 000021 (¶38), 000022 (¶44).

190. A POSITA would have had a reasonable expectation of success when combining Rusert and Zheng. As explained above, the combination applies teachings from each reference according to their known purposes, in a conventional manner as taught by Rusert and Zheng, without changing their principle of operation. Furthermore, the teachings are complementary because both references teach aspects of block-based video encoding, one from an H.264 perspective and the other from an H.265 perspective. *E.g.* Ex-1004, ¶25, ¶39;

Ex-1005, ¶¶3-4, ¶28, ¶96, ¶129; Ex-1055, 000014 (¶¶2-3), 000016 (¶19),000025 (¶55), 000031 (¶78); *see also* Ex-1011, ¶3, ¶5, ¶38. Zheng's teachings apply a basic concept to further Rusert's stated goal of reducing the number of PMV candidates, by eliminating redundant ones as explained above. *E.g.*, Ex-1004, ¶25, ¶39, ¶¶42-44; Ex-1005, ¶50, ¶¶62-63; Ex-1055, 000019 (¶34), 000020-21 (¶¶37-38).

191. Additionally, the combination would have had predictable results, as explained above, as the combination does not modify Rusert or Zheng in a way that would render any reference inoperative. To the contrary, the similarities of the architectures and video standards would have given a POSITA a reasonable expectation of success in combining their teachings. Ex-1004, ¶67, ¶116; Ex-1005, ¶¶3-4, ¶28, ¶96, ¶129; Ex-1055, 000014 (¶¶2-3), 000016 (¶19), 000025 (¶55), 000031 (¶78).

192. Given the education and experience of a POSITA (*supra* §III), a POSITA would have been more than capable of applying Zheng's teachings to Rusert because it would simply have applied the concept of excluding candidates in a particular scenario as taught by Zheng. Motion estimation and the use of predicted motion vectors had been widespread for many years. *Supra* §I. These were basic aspects of video encoding/decoding that a POSITA would have been knowledgeable about.

193. Based on the foregoing explanations, Ground 1 teaches claim 11.

6. Dependent Claim 12

[12pre]	**The method according to claim 9 further comprising**

194. As I explained above, Ground 1 teaches the method of claim 9. *Supra* §V.A.3.

[12a]	**determining a maximum number of spatial motion vector prediction candidates to be included in a merge list; and**

195. Ground 1 teaches limitation [12a], as explained below. The spatial motion vector prediction candidates (e.g., PMV candidates in Rusert and merge candidates in Zheng) and merge list (e.g., PMV_CANDS in Rusert and merge candidate set in Zheng) were explained for claim 9. *Supra* §V.A.3[9a].

196. Rusert teaches **determining a maximum number of spatial motion vector prediction candidates to be included in the merge list** (e.g., PMV_CANDS). For example, Rusert teaches "the number of candidates in PMV_CANDS *may be limited* to a pre-defined or dynamically obtained number." Ex-1004, ¶73, ¶¶84-90. As some examples, Rusert teaches "a maximum of four candidates in the list" and "us[ing] seven" for a maximum. Ex-1004, ¶107. Rusert also teaches that "it is possible that PMV_CANDS list size is set to one, such that no bits need to be sent for index signaling." Ex-1004, ¶77.

197. Rusert teaches several benefits to determining a maximum number of candidates for PMV_CANDS. For example, "[l]imiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words." Ex-1004, ¶84, ¶70, ¶13 ("The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use[.]"). With "a bigger maximum,… the number of bits needed to specify the candidate vector also increases" and "many vectors can be represented using several candidates, which is unnecessary." Ex-1004, ¶107. By determining a maximum number of candidates, Rusert balances between "chances increas[ing] that a suitable vector can be found" and "redundant representation" that "grows the more vectors are added." Ex-1004, ¶107. In short, Rusert teaches determining a maximum number of candidates for PMV_CANDS, e.g., 4 candidates, to balance the increased chance of a suitable match with the increased cost of using longer code words. *Id.*

198. Rusert teaches variable-length coding ("VLC") examples for index values which vary based on the "Maximum list size C" of PMV_CANDS. Ex-1004, ¶88 (Table 1). "The VLC table used can depend on the maximum number of candidates in PMV_[C]ANDS (the list size), as e.g. dynamically adapted according to the methods above." Ex-1004, ¶88. "Table 1 below presents

some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list." *Id.* Therefore, Rusert teaches determining a maximum number of candidates to be included in PMV_CANDS, e.g. as denoted by "C," which dictates the encoding table for index values and controls the maximum number of candidates in the PMV_CANDS list. *Id.*

199. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Zheng's teachings of the merge mode, apply to a merge list. Thus, the combination of Rusert and Zheng teaches determining a maximum number of spatial motion vector prediction candidates to be included in a merge list.

200. Zheng provides a further motivation to do so because it teaches that in order to enable parallel encoding that the merge candidates for a specific PU of a partitioned CU would be limited by "remov[ing] merge candidates from the merge candidate set if the merge candidate is located inside another PU of the same CU."

Ex-1005, ¶69, Fig. 3A; Ex-1055, 000004 (Fig. 3A), 000021-22 (¶¶41-43). Zheng specifically notes that an encoding process that "requires that all PUs have been encoded/decoded prior to developing the merge candidate set of subsequent PUs" restricts and results in "the ability to process multiple PUs of one CU in parallel is limited" because "the encoder/decoder must determine the final motion information of all previous PUs in the same CU before constructing the merge candidate set of a current PU." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng concludes that "the comparison operation for each candidate block may increase the computational complexity for the encoder/decoder." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng's proposal suggests that a maximum number of merge candidates are considered for each PU of the same CU, that is dependent on the location of the PU within the CU to take into account overlap of the merge candidates. Therefore, a POSITA would have found it obvious that the teachings of determining a maximum number of spatial motion vector prediction candidates apply to a merge list.

201. Based on the foregoing explanations, Ground 1 teaches limitation [12a].

[12b]	**limiting the number of spatial motion vector prediction candidates in the merge list** [to be] **smaller or equal to the maximum number.**

202. Ground 1 teaches limitation [12b], as explained below.

203. Rusert teaches **limiting the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number**. For example, Rusert teaches "an outwards going scan… to obtain motion vectors to update PMV_[C]ANDS… may be terminated… as soon as a pre-defined number of unique PMV candidates have been found[.]" Ex-1004, ¶¶44-48. By terminating the scan "as soon as a pre-defined number of unique PMV candidates have been found," the number of candidates in PMV_CANDS is limited to the pre-defined number, which is the maximum number, because no additional candidates are considered for inclusion in PMV_CANDS. *Id*.

204. In addition, Rusert teaches that "the candidate at the end of the PMV_CANDS list may be removed" in order to limit "the number of candidates in PMV_CANDS… to a pre-defined… number. Ex-1004, ¶73. Thus, the number of candidates in PMV_CANDS is smaller or equal to the maximum number because candidates are removed in order to limit the number of candidates to the maximum number. Moreover, by teaching a "maximum" number of candidates in the PMV_CANDS list, Rusert teaches a limit (the maximum) to the number of candidates in the list.

205. Rusert further teaches VLC examples based on the "Maximum list size C" of PMV_CANDS. Ex-1004, ¶88 (Table 1). "The VLC table used can depend on the maximum number of candidates in PMV_[C]ANDS (the list size),

as e.g. dynamically adapted according to the methods above." Ex-1004, ¶88. "Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list." *Id.* Therefore, Rusert teaches limiting the number of candidates in PMV_CANDS to be smaller or equal to the maximum number, e.g. as denoted by "C," in accordance with the encoding tables in Table 1. *Id.* For example, when the maximum list size C is 4, Table 1 lists encodings for four index values to encode up to four values in PMV_CANDS. None of the encodings exceed the maximum list size C.

206. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Zheng's teachings of the merge mode, apply to a merge list. Thus, the combination of Rusert and Zheng teaches limiting the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number.

207. Zheng provides a further motivation to do so because it teaches that in order to enable parallel encoding that the merge candidates for a specific PU of a partitioned CU would be limited by "remov[ing] merge candidates from the merge candidate set if the merge candidate is located inside another PU of the same CU." Ex-1005, ¶69, Fig. 3A; Ex-1055, 000004 (Fig. 3A), 000021-22 (¶¶41-43). Zheng specifically notes that an encoding process that "requires that all PUs have been encoded/decoded prior to developing the merge candidate set of subsequent PUs" restricts and results in "the ability to process multiple PUs of one CU in parallel is limited" because "the encoder/decoder must determine the final motion information of all previous PUs in the same CU before constructing the merge candidate set of a current PU." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng concludes that "the comparison operation for each candidate block may increase the computational complexity for the encoder/decoder." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng's proposal suggests that a maximum number of merge candidates are considered for each PU of the same CU, that is dependent on the location of the PU within the CU to take into account overlap of the merge candidates. Therefore, a POSITA would have found it obvious that the teachings of limiting the number of spatial motion vector prediction candidates to be smaller or equal to the maximum number apply to a merge list.

208. Based on the foregoing explanations, Ground 1 teaches limitation [12b].

7. Dependent Claim 13

[13pre] The method according to claim 12 comprising:

209. As I explained above, Ground 1 teaches the method of claim 12.

[13a] examining, if the number of spatial motion vector prediction candidates in the merge list [is] **smaller than the maximum number;**

210. Ground 1 teaches limitation [13a], as explained below.

211. Rusert teaches **examining, if the number of spatial motion vector prediction candidates in the merge list [is] smaller than the maximum number**. As I explained above with respect to claim 12, Rusert teaches determining a maximum number of candidates for PMV_CANDS and limiting the number of candidates in PMV_CANDS to the maximum number. Ex-1004, ¶13, ¶¶44-48, ¶70, ¶73, ¶77, ¶¶84-90, ¶107; *supra* §V.A.6 (claim 12). For example, Rusert examines if the number of candidates in PMV_CANDS is smaller than the maximum number when determining whether to continue its scan for additional candidates. Rusert teaches "terminat[ing]" a scan for new candidates "as soon as a pre-defined number of unique PMV candidates have been found[.]" Ex-1004, ¶¶44-48. Here, "an outwards going scan… around the current block" is performed to obtain candidates "to update PMV_[C]ANDS[,]" and the number of candidates

in PMV_CANDS is examined to determine whether to continue or terminate the scan. *Id*. If the number of candidates is less than the maximum number, or the "pre-defined number of unique PMV candidates[,]" then the scan continues; but if the number of candidates is equal to the maximum number, then the scan is "terminated[.]" *Id*.

212. In addition, Rusert teaches that "the candidate at the end of the PMV_CANDS list may be removed" in order to limit "the number of candidates in PMV_CANDS… to a pre-defined… number." Ex-1004, ¶73. Thus, Rusert teaches examining if the number of candidates in PMV_CANDS is smaller than the maximum number to determine whether to remove a candidate for another candidate or to add the other candidate.

213. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Lim's teachings of the merge mode, apply to a merge list. Thus, the combination of Rusert and Zheng teaches examining if the number of spatial motion vector prediction candidates in the merge list is smaller than the maximum number.

214. Zheng provides a further motivation to do so because it teaches that in order to enable parallel encoding that the merge candidates for a specific PU of a partitioned CU would be limited by "remov[ing] merge candidates from the merge candidate set if the merge candidate is located inside another PU of the same CU." Ex-1005, ¶69, Fig. 3A; Ex-1055, 000004 (Fig. 3A), 000021-22 (¶¶41-43). Zheng specifically notes that an encoding process that "requires that all PUs have been encoded/decoded prior to developing the merge candidate set of subsequent PUs" restricts and results in "the ability to process multiple PUs of one CY in parallel is limited" because "the encoder/decoder must determine the final motion information of all previous PUs in the same CU before constructing the merge candidate set of a current PU." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng concludes that "the comparison operation for each candidate block may increase the computational complexity for the encoder/decoder." Ex-1005, ¶62; Ex-1055, 000020-21 (¶37). Zheng's proposal suggests that a maximum number of merge candidates are considered for each PU of the same CU, that is dependent on the location of the PU within the CU to take into account overlap of the merge candidates. Therefore, a POSITA would have found it obvious that the teachings of examining if the number of spatial motion vector prediction candidates is smaller than the maximum number apply to a merge list.

215. Based on the foregoing explanations, Ground 1 teaches limitation [13a].

[13b]	**if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;**

216. Ground 1 teaches limitation [13b], as explained below.

217. Ground 1 teaches **examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction.** While scanning blocks for new candidates, Rusert examines whether the block to which a potential candidate belongs is available for motion prediction because motion vectors are not available for some blocks. For example, Rusert teaches three exemplary reasons why a block would be unavailable for motion prediction, any one of those teachings satisfies this limitation.

218. First, Rusert teaches that blocks that have "no motion vector present" or have "the same [motion vector] as a block earlier in the sequence" are not available. Ex-1004, ¶54. Blocks that have no motion vector present do not have motion information for motion prediction and cannot provide a motion vector candidate. Since Rusert seeks to avoid duplicates, blocks that have the same motion vector as a block earlier in the sequence are not available for motion prediction.

219. Second, Rusert teaches that blocks "coded after the present block[]" "would never be available[.]" Ex-1004, ¶54. For example, blocks below and to the right of the current block would be coded after the present block. *Id.* Quite simply, these blocks are never available because they have not been coded yet and therefore do not have any motion information for motion prediction. No motion vector has been assigned to those blocks yet and therefore the blocks cannot provide a motion vector candidate.

220. Third, blocks are "sometimes… available depending upon the traversal pattern used[.]" Ex-1004, ¶54. Depending on the pattern used to reach the current block, some blocks would not be coded yet, and similar to the first condition explained above, these blocks do not have any motion information for motion prediction.

221. Thus, with any of these considerations, Rusert teaches examining whether a block is available for motion prediction.

222. In the context of Rusert's nomenclature, Rusert's block is a prediction unit because it is the unit for which a motion vector is assigned for motion prediction. Ex-1004, ¶¶2-5, ¶36, ¶43; *supra* §V.A.3[9a] (explaining PUs and Rusert's blocks). Therefore, Rusert teaches this limitation.

223. Additionally, Ground 1 applies Rusert's motion vector teachings to Prediction Units. *Supra* §V.A.3[9a] (explaining application of block and PU

teachings and motivation to combine Rusert and Zheng); Ex-1004, ¶¶3-4, ¶36; Ex-1005, ¶¶3, 27-29, 38; Ex-1055, 000014 (¶2), 000016-17 (¶¶19-22), 000017 (¶24). As Zheng explains, a "prediction unit"—a term used in H.265—was a type of block, and motion vectors were assigned to blocks, including PUs. Ex-1005, ¶¶29, 36, 38; Ex-1055, 000016 (¶20), 000016-17 (¶22), 000017 (¶24); *supra* §I. Therefore, it would have been obvious to apply Rusert's motion vector teachings to PUs, and Ground 1 teaches examining whether the prediction unit, to which the potential spatial motion vector prediction candidate belongs, is available for motion prediction.

224. Furthermore, this would have been obvious based on the teachings of Rusert and Zheng. As explained above, Rusert teaches three reasons for examining whether a block is available for motion prediction as part of its process for finding and evaluating potential spatial motion vector prediction candidates. Zheng teaches that, in the H.265 terminology, the block for which motion vectors are assigned is called a PU. Ex-1005, ¶¶28-30, 38; Ex-1055, 000014 (¶2), 000016-17 (¶¶19-22), 000017 (¶24). Therefore, when scanning for potential spatial motion vector prediction candidates as Rusert teaches, it would have been obvious to scan PUs in the H.265 context, and it would have been obvious to examine whether the PU is available for motion prediction, using the various criteria Rusert teaches.

225. Based on the foregoing explanations, Ground 1 teaches limitation [13b].

[13c]	**if so, performing at least one of the following:**
[13d]	**for a potential spatial motion vector prediction candidate on a left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13e]	**the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;**
[13f]	**the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;**
[13g]	**for a potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13h]	**the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;**
[13i]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;**
[13j]	**for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**

[13k]	**for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and**
[13l]	**for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13m]	**all the other potential spatial motion vector prediction candidates have been included in the merge list;**
[13n]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**
[13o]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.**

226. Limitation [13c] requires "performing at least one of the following." Therefore, I understand that this limitation is satisfied if the following claim requirement (limitation [13k]) is satisfied:

> **for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;**

227. Ground 1 teaches this limitation, as explained below. As an initial matter, in the context of Rusert's nomenclature, Rusert's block is a prediction unit because it is the unit for which a motion vector is assigned for motion prediction. Ex-1004, ¶¶2-5, ¶36, ¶43. Additionally, the combination of Ground 1 applies Rusert's motion vector teachings to Prediction Units. *Supra* §V.A.3[9a] (explaining application of block and PU teachings and motivation to combine Rusert and Zheng); Ex-1004, ¶¶3-4, ¶36; Ex-1005, ¶¶33-36, ¶¶64-67; Ex-1055, 000016-17 (¶22), 000021 (¶39), 000031 (¶79), 000032 (¶84). A POSITA would have understood that a "prediction unit," which is a terminology introduced in H.265, was a type of block, and motion vectors were assigned to PUs. Ex-1005, ¶¶28-30, 38; Ex-1055, 000014 (¶2), 000016 (¶¶19-21), 000017 (¶24); *see also* Ex-1011, ¶¶33-36; *supra* §I. Therefore, when Rusert teaches its scanning and analysis process for motion vectors from various blocks, it would have been obvious to apply those teachings to PUs in the H.265 context because motion vectors were assigned on a PU basis in H.265. *Id.*

228. If Rusert's scan continues (the number of candidates in PMV_CANDS is less than a pre-defined number), Ground 1 applies the below-explained teachings **for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit**. For example, Rusert teaches a scan pattern in

Fig. 3n, illustrated below, that "has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance." Ex-1004, ¶66. As illustrated in Fig. 3n, the current block is "." (highlighted red below); the scan starts with a candidate above the current block (e.g., block "1"), then scans a candidate left of the current block (e.g., block "2") second, and so on, progressing through blocks "3," "4," and then "5." Ex-1004, ¶¶65-66, Fig. 3n. The candidate for block "5" (highlighted green below) is below the candidate for block "2" (highlighted blue below), which is on the left side of the current block (in red). *Id.*:

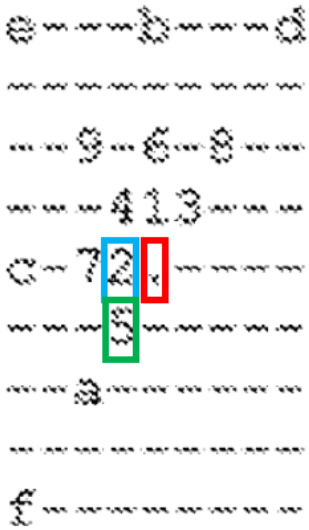


Fig. 3n

229. Therefore, Rusert analyzes a potential spatial motion vector prediction candidate (e.g., the candidate from block "5" highlighted in green) which is below the potential candidate on the left (e.g., from block "2" highlighted in blue) of the

current block (e.g., block "." highlighted in red) for which candidates are being evaluated. Ex-1004, ¶66, Fig. 3n. As explained above, these blocks are prediction units, and Ground 1 applies these teachings to PUs in the H.265 context.

230. Ground 1 teaches **excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.** For example, when analyzing a new candidate, Rusert teaches "comparing the [PMV] candidates already in [PMV_CANDS] with the new [PMV candidate] that could be added, and if a duplicate is found, … skipping the new [PMV candidate]." Ex-1004, ¶¶71-72, ¶21, ¶62 ("It has been identified as advantageous to discard such duplicates.")

231. Rusert analyzes new candidates following a scan order of nearby blocks, including the pattern of Fig. 3n illustrated below. Block 1 is analyzed first, then blocks 2, 3, and 4 in that order, adding candidates from those blocks to PMV_CANDS. When the 5th block is analyzed (highlighted green), Rusert will already have analyzed blocks 1, 2, 3, and 4. Ex-1004, ¶44, ¶¶65-66, Fig. 3n. Therefore, when Rusert evaluates whether to update PMV_CANDS with the spatial motion vector prediction candidate from block 5, Rusert compares the new candidate with the candidates already in PMV_CANDS, e.g., the candidates from

blocks 1, 2, 3, and 4, to determine if it is a duplicate or within a threshold distance (meaning it is within a similarity measure) of an existing candidate in PMV_CANDS. Ex-1004, ¶¶71-72. This comparison includes a comparison of the candidate from block 5 with the candidate from block 2, and Rusert excludes the potential spatial motion vector prediction candidate for block 5 from PMV_CANDS if it is a duplicate or within a similarity threshold as compared to the candidate on the left side of the PU (block 2). *Id.* In short, for the scan pattern of Fig. 3n, Rusert teaches excluding the candidate below the candidate to the left of the current block from PMV_CANDS if it is essentially similar (e.g., a duplicate or within a threshold distance) compared to the candidate to the left. Ex-1004, Fig. 3n:

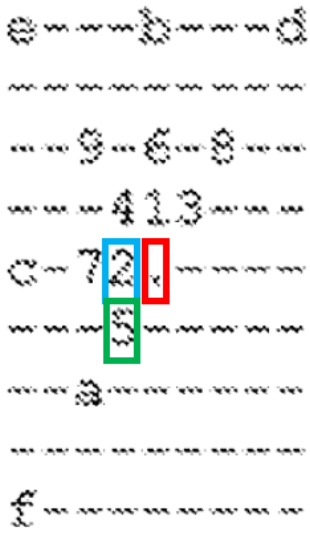


Fig. 3n

232. Rusert looks to whether the candidates are essentially similar by looking at whether they are duplicates or within a threshold distance. Indeed, Rusert uses its threshold comparison to "skip[] new motion vectors … that are similar but not equal, such as pairs of motion vectors *that have a similarity measure* smaller than a pre-defined threshold …" Ex-1004, ¶72. Therefore Rusert looks to whether candidates are essentially similar. *Id.*

233. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Zheng's teachings of the merge mode, apply to a merge list. Thus, the combination of Rusert and Zheng excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.

234. Based on the foregoing explanations, Ground 1 teaches limitations [13c]-[13o].

8. Dependent Claim 14

14.	**The method according to claim 9 comprising selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.**

235. Ground 1 teaches claim 14, as explained below.

236. As I have explained above, the combination of Rusert and Zheng teaches claim 9.

237. Rusert further teaches **selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the encoded block of pixels**. For example, Rusert teaches "signaling which PMV [candidate] is used for motion vector prediction[.]" Ex-1004, ¶84; *supra* §§V.A.3[9a], [9c], [9g]. "[A] code 'index'… is sent to *select* a particular PMV candidate… from a list of PMV candidates, PMV_CANDS" and the particular candidate is used to "reconstruct MV=DMV+PMV" for the current block of pixels. Ex-1004, ¶¶36-37. Therefore, a candidate is selected from PMV_CANDS using an index, and the selected candidate represents the "predicted motion vector (PMV)" used to reconstruct the motion vector for a block. *Id.*, ¶36. Therefore, Rusert teaches selecting one motion vector prediction candidate from PMV_CANDS to represent a motion vector prediction for the encoded block of pixels.

238. Additionally, Rusert teaches exemplary codes to signal the index of the selected candidate from PMV_CANDS. Ex-1004, ¶¶88-102 (exemplary codes

for index values to signal which candidate from PMV_CANDS is used for motion information). For example, Table 1 includes codes for lists of various sizes including codes to indicate the first, second or third candidate in a list size of 3 candidates. *Id.,* ¶88.

239. Rusert teaches that the techniques of selecting a candidate using an index are used for decoding an encoded block of pixels. For example, Rusert teaches that its teachings apply to "video decoding, wherein the current block is the block being… decoded[.]" Ex-1004, ¶23, ¶35 ("The methods disclosed herein are performed… in the decoder during decoding."). The current block is part of an "encoded video stream" that is "passe[d] to a decoder… employed in decoding the encoded video stream." Ex-1004, ¶34. The index that identifies a particular candidate is also transmitted to the decoder as the motion information for the encoded block. Ex-1004, Fig. 2b. The decoder thus selects a candidate from PMV_CANDS using the transmitted index, and the selected candidate represents the "predicted motion vector (PMV)" used to reconstruct the motion vector for the encoded block. Ex-1004, ¶36. Therefore, Rusert teaches a decoder using an index to select a candidate from PMV_CANDS to represent a motion vector prediction for the encoded block of pixels. *See supra* §V.A.3[9g].

240. **Combining Rusert and Zheng**. As I explained above, a POSITA would have been motivated to combine Rusert and Zheng, where the PMV

candidates taught in Rusert are merge candidates and the PMV_CANDS list taught in Rusert functions as a merge list. *See, e.g., supra* §§V.A.3[9a], [9c]. Accordingly, a POSITA would have understood that the Rusert's teachings above with respect to PMV_CANDS, in view of Zheng's teachings of the merge mode, apply to a merge list. Thus, the combination of Rusert and Zheng teaches selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.

241. Based on the foregoing explanations, Ground 1 teaches claim 14.

9. Independent Claim 16

[16pre] An apparatus comprising a processor and a memory including computer program code, the memory and the computer program code configured to, with the processor, cause the apparatus to:

242. To the extent that the preamble is limiting, Ground 1 teaches limitation [16pre], as explained below.

243. Rusert teaches an **apparatus comprising a processor and a memory including computer program code, the memory and computer program code is configured to, with the processor, cause the apparatus to.** For example, Rusert teaches "[t]he present application relates to … a video encoding apparatus, a video decoding apparatus, and a computer-readable medium." Ex-1004, ¶1; *see also* Ex-1004, ¶¶24, 26-27, 114, 116, claim 17, claim 18. The "video encoding apparatus compris[es] a processor" and that the "video decoding apparatus

compris[es] a processor[.]" Ex-1004, ¶¶24-25. Rusert further teaches "a computer-readable medium, carrying instructions, which when executed by computer logic[,]" such as a processor, "causes said computer logic to carry out any of the methods disclosed herein." Ex-1004, ¶26. Therefore, Rusert teaches an "apparatus comprising a processor" and memory "carrying instructions" that cause the apparatus to "carry out any of the methods disclosed herein." Ex-1004, ¶¶24-26. At the very least, it would have been obvious to a POSITA for a video encoder and a video decoder to include a processor and a memory, such as a hard drive, to execute software because conventional computers have included processors and memory for this purpose for decades.

244. Based on the foregoing explanations, Ground 1 teaches the preamble of claim 16.

[16a]	**receive an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;**

245. Ground 1 teaches limitation [16a]. Limitation [16a] is identical to limitation [9a], except that limitation [16a] recites "receive" while limitation [9a] recites "receiving." For the same reasons I have discussed for limitation [9a], Ground 1 teaches limitation [16a]. *Supra* §V.A.3[9a].

[16b]	**determine a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

246. Ground 1 teaches limitation [16b]. Limitation [16b] is identical to limitation [9b], except that limitation [16b] recites "determine" while limitation [9b] recites "determining." For the same reasons I have discussed for limitation [9b], Ground 1 teaches limitation [16b]. *Supra* §V.A.3[9b].

[16c]	**select a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

247. Ground 1 teaches limitation [16c]. Limitation [16c] is identical to limitation [9c], except that limitation [16c] recites "select" while limitation [9c] recites "selecting." For the same reasons I have discussed for limitation [9c], Ground 1 teaches limitation [16c]. *Supra* §V.A.3[9c].

[16d]	**determine a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

248. Ground 1 teaches limitation [16d]. Limitation [16d] is identical to limitation [9d], except that limitation [16d] recites "determine" while limitation [9d] recites "determining." For the same reasons I have discussed for limitation [9d], Ground 1 teaches limitation [16d]. *Supra* §V.A.3[9d].

[16e]	**compare motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;**

249. Ground 1 teaches limitation [16e], as explained below.

250. Limitation [16e] is substantially similar to limitation [9e]. Their differences are shown below:

[16e]	[9e]
compare motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;	comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

251. As an initial matter, "*the* spatial motion vector prediction candidate" of limitation [16e] lacks antecedent basis. This limitation is rendered obvious for the same reasons that "*another* spatial motion vector prediction candidate" of

limitation [9e] is obvious because both limitations reference a spatial motion vector prediction candidate other than the first spatial motion vector prediction candidate.

252. Furthermore, "*in* the *determined subset*" is rendered obvious for the same reasons that "*of* the *[sub]set*" is obvious because a comparison between two spatial motion vector prediction candidates in a subset teaches a comparison of two spatial motion vector prediction candidates in a subset of the set. Here, the "determined subset" references "a subset" in limitation [16d] which is a subset of "a set" in limitation [16b]. Likewise, "the subset" of limitation [9c] is "a set" in limitation [9b].

253. At the very least, it would be obvious that Ground 1 teaches "compare … with motion information of the spatial motion vector prediction candidate in the determined subset," because Rusert teaches, for example, comparing the selected PMV candidate with PMV candidates currently in PMV_CANDS. *See* §V.A.3[9e]. For the reasons I explained for limitation [9e], Ground 1 teaches limitation [16e]. *Supra* §V.A.3[9e].

[16f]	**exclude the first spatial motion vector prediction candidate from the merge list, if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other; and**

254. Ground 1 teaches limitation [16f], as explained below.

255. Limitation [16f] is substantially similar to limitation [9f]. Their differences are shown below:

[16f]	[9f]
exclude the first spatial motion vector prediction candidate from the merge list, if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other; and	if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and

256. Limitation [16f] recites the same limitations as limitation [9f] except in a different order. Limitation [16f] recites "exclude … from the merge list" before "if at least one of the comparisons … correspond with each other" while limitation [9f] recites "excluding … from the merge list" after "if at least one of the comparisons … correspond with each other." This is not a meaningful difference because both limitation [16f] and limitation [9f] require excluding the first spatial motion vector prediction candidate from the merge list if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other. For the same reasons I have discussed for limitation [9f], Ground 1 teaches limitation [16f]. *Supra* §V.A.3[9f].

[16g]	**select a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

257. Ground 1 teaches limitation [16g]. Limitation [16g] is identical to limitation [9g], except that limitation [16g] recites "select" while limitation [9g] recites "selecting." For the same reasons I have discussed for limitation [9g], Ground 1 teaches limitation [16g]. *Supra* §V.A.3[9g].

10. Independent Claim 18

[18pre]	**A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprise instructions for use by an encoder, said program code comprises instructions for:**

258. To the extent that the preamble is limiting, Ground 1 teaches limitation [18pre], as explained below.

259. As an initial matter, limitations [18pre] and limitation [18a] recite "an encoder." This appears to be a typographical error arising from copying the preamble from claim 17. *Compare* Ex-1001, claim 18 *with* Ex-1001, claim 17. While limitation [18pre] states the limitations of claim 18 are for "an encoder", the limitations are inconsistent with an encoder because the limitations recite, for example, "receiving an encoded block of pixels" (limitation [18a]) and "decoding the encoded block of pixels" (limitation [18g]). Thus, it is my opinion that

limitation [18pre] would be understood to be "for use by a decoder" not "for use by an encoder[.]" Regardless, Rusert teaches both "a video encoding apparatus" and "a video decoding apparatus[.]" Ex-1004, ¶1.

260. Rusert teaches **a non-transitory computer readable medium having stored thereon a computer executable program code for use by [a decoder], said program codes comprising instructions for**. For example, Rusert teaches "a computer-readable medium, carrying instructions, which, when executed by computer logic, causes said computer logic to carry out any of the methods disclosed herein." Ex-1004, ¶26, ¶1 ("The present application relates to… a video encoding apparatus, a video decoding apparatus, and a computer-readable medium."), claim 19. The computer-readable medium is used, for example, by "a video encoding apparatus" and "a video decoding apparatus[.]" Ex-1004, ¶¶24-25. At the very least, it would have been obvious to a POSITA for a video encoder and a video decoder to include a non-transitory computer-readable medium, such as a hard drive, that stores computer executable program code, such as software, because conventional computers have included hard drives storing software for decades.

261. Based on the foregoing explanations, Ground 1 teaches the preamble of claim 18.

[18a]	**receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;**

262. Ground 1 teaches limitation [18a]. Limitation [18a] is identical to limitation [9a]. For the same reasons I have discussed for limitation [9a], Ground 1 teaches limitation [18a]. *Supra* §V.A.3[9a].

[18b]	**determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

263. Ground 1 teaches limitation [18b]. Limitation [18b] is identical to limitation [9b]. For the same reasons I have discussed for limitation [9b], Ground 1 teaches limitation [18b]. *Supra* §V.A.3[9b].

[18c]	**selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

264. Ground 1 teaches limitation [18c]. Limitation [18c] is identical to limitation [9c]. For the same reasons I have discussed for limitation [9c], Ground 1 teaches limitation [18c]. *Supra* §V.A.3[9c].

[18d]	**determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

265. Ground 1 teaches limitation [18d]. Limitation [18d] is identical to limitation [9d]. For the same reasons I have discussed for limitation [9d], Ground 1 teaches limitation [18d]. *Supra* §V.A.3[9d].

[18e]	**comparing motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;**

266. Ground 1 teaches limitation [18e], as explained below.

267. Limitation [18e] is substantially similar to limitation [9e]. Their differences are shown below:

[18e]	[9e]
comparing motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;	comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

268. As an initial matter, "*the* spatial motion vector prediction candidate" of limitation [18e] lacks antecedent basis. This limitation is rendered obvious for the same reasons that "*another* spatial motion vector prediction candidate" of limitation [9e] is obvious because both limitations reference a spatial motion vector prediction candidate other than the first spatial motion vector prediction candidate.

269. Furthermore, "*in* the *determined subset*" is rendered obvious for the same reasons that "*of* the *[sub]set*" (incorporating the certificate of correction) is obvious because a comparison between two spatial motion vector prediction candidates in a set teaches a comparison of two spatial motion vector prediction candidates in a subset of the set. Here, the "determined subset" references "a subset" in limitation [18d] which is a subset of "a set" in limitation [18b]. Likewise, "the [sub]set" of limitation [9c] is subset of "a set" in limitation [9b].

270. At the very least, it would be obvious that Ground 1 teaches "comparing … with motion information of the spatial motion vector prediction candidate in the determined subset," because Rusert teaches, for example, comparing the selected PMV candidate with PMV candidates currently in PMV_CANDS. *See* §V.A.3[9e]. For the reasons I explained for limitation [9e], Ground 1 teaches limitation [18e]. *Supra* §V.A.3[9e].

[18f]	**if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and**

271. Ground 1 teaches limitation [18f]. Limitation [18f] is identical to limitation [9f]. For the same reasons I have discussed for limitation [9f], Ground 1 teaches limitation [18f]. *Supra* §V.A.3[9f].

[18g]	**selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

272. Ground 1 teaches limitation [18g]. Limitation [18g] is identical to limitation [9g]. For the same reasons I have discussed for limitation [9g], Ground 1 teaches limitation [18g]. *Supra* §V.A.3[9g].

11. Dependent Claim 26

26.	**The apparatus according to claim 16 wherein the is apparatus is further caused to compare motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.**

273. Ground 1 teaches claim 26. Claim 26 is identical to claim 10, except that claim 10 recites "The method according to claim 9 comprising comparing …," whereas claim 26 recites "The apparatus according to claim 16 wherein the apparatus is further caused to compare …" I addressed the apparatus of claim 16 above. *Supra* §V.A.9. The rest of claim 26 is satisfied for the same reasons I explained for claim 10. *Supra* §V.A.4.

12. Dependent Claim 27

27.	**The apparatus according to claim 16 wherein the apparatus is further caused to examine whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, exclude the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.**

274. Ground 1 teaches claim 27. Claim 27 is identical to claim 11, except that claim 11 recites "The method according to claim 9 comprising examining …" and "excluding," whereas claim 27 recites "The apparatus according to claim 16 wherein the apparatus is further caused to examine …" and "exclude." I addressed the apparatus of claim 16 above. *Supra* §V.A.9. The rest of claim 27 is satisfied for the same reasons I explained for claim 11. *Supra* §V.A.5.

13. Dependent Claim 28

[28pre]	**The apparatus according to claim 16 wherein the apparatus is further caused to:**
[28a]	**determine a maximum number of spatial motion vector prediction candidates to be included in a merge list; and**
[28b]	**limit the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number.**

275. Ground 1 teaches claim 28. Claim 28 is identical to claim 12, except that claim 12 recites "The method according to claim 9 further comprising …," followed by limitations using verbs in their gerund form, whereas claim 28 recites "The apparatus according to claim 16 wherein the apparatus is further caused to …," followed by the same limitations with the verbs in their base form. I addressed the apparatus of claim 16 above. *Supra* §V.A.9. The rest of claim 28 is satisfied for the same reasons I explained for claim 12. *Supra* §V.A.6.

14. Dependent Claim 29

[29pre]	**The apparatus according to claim 28 wherein the apparatus is further caused to:**
[29a]	**examine if the number of spatial motion vector prediction candidates in the merge list smaller than the maximum number;**
[29b]	**if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;**
[29c]	**if so, perform at least one of the following:**
[29d]	**for a potential spatial motion vector prediction candidate on a left side of the prediction unit, exclude the potential spatial motion vector**

prediction candidate from the merge list if any of the following conditions are fulfilled:

[29e] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[29f] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;

[29g] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29h] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;

[29i] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;

[29j] for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[29k] for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and

[29l]	**for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[29m]	**all the other potential spatial motion vector prediction candidates have been included in the merge list;**
[29n]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**
[29o]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.**

276. Ground 1 teaches claim 29. Claim 29 is identical to claim 13, except that claim 13 recites "The method according to claim 12 comprising …," followed by limitations using verbs in their gerund form, whereas claim 29 recites "[t]he apparatus according to claim 28 wherein the apparatus is further caused to …," followed by the same limitations with the verbs in their base form. I addressed the apparatus of claim 28 above. *Supra* §V.A.13. The rest of claim 29 is satisfied for the same reasons I explained for claim 13. *Supra* §V.A.7.

15. Dependent Claim 30

30.	**The apparatus according to claim 16 wherein the apparatus is further caused to select one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.**

277. Ground 1 teaches claim 30. Claim 30 is identical to claim 14, except that claim 14 recites "The method according to claim 9 comprising selecting …,"

whereas claim 30 recites "The apparatus according to claim 16 wherein the apparatus is further caused to select …" I addressed the apparatus of claim 16 above. *Supra* §V.A.9. The rest of claim 30 is satisfied for the same reasons I explained for claim 14. *Supra* §V.A.8.

B. Ground 2

278. It is my opinion that a POSITA would have found the Challenged Claims obvious based on Ground 2, i.e., the teachings of Nakamura and WD4. Nakamura teaches and suggests the limitations of the challenged claims. WD4 provides additional teachings on limitations [9a] and [9g].

1. Nakamura (Ex-1007, Ex-1008, Ex-1009) and WD4 (Ex-1010)

279. Nakamura is an HEVC proposal (numbered JCTVC-F419) presented at the 6th meeting of the Joint Collaborative Team on Video Coding (JCT-VC), held between July 14-22, 2011, in Torino, Italy. Nakamura is titled "Unification of derivation process for merge mode and MVP" and authored by Hiroya Nakamura et al. The Nakamura proposal comprises 6 files, which were part of the same proposal and uploaded to the JCT-VC website in one zip archive file. The 6 files in Nakamura include a main document describing the proposed techniques of the proposal (Ex-1007), a Working Draft description of the proposed techniques of the proposal (Ex-1008), and a presentation slide deck illustrating the proposed techniques of the proposal (Ex-1009).

280. WD4 is version 3 of the Working Draft 4 of the HEVC standard (H.265) developed by the JCT-VC. WD4 was the output of the 6th JCT-VC meeting held July 14-22, 2011, in Torino, Italy. Ex-1010, 000001.

281. I understand that the Nakamura proposal and WD4 are prior art to the '833 patent under at least pre-AIA §102(a). Nakamura was publicly available by at least the time of the 6th JCT-VC meeting on July 22, 2011. Ex-1007, 000001; Ex-1014, ¶15, ¶36; Ex-1049. The input to the 6th meeting was Working Draft 3 ("WD3"), and the output was Working Draft 4. At that meeting, attendees drafted Working Draft 4, which was made publicly available through the JCT-VC website shortly after the meeting; version 3 of Working Draft 4 ("WD4") was publicly available through the JCT-VC website by September 8, 2011. Ex-1010, 000001; Ex-1014, ¶15, ¶37; Ex-1049. Nakamura and WD4 were made available to all meeting participants and were publicly accessible on the JCT-VC's website. *See generally* Ex-1014, Ex-1049. Anyone with Internet access could download Nakamura and WD4 from the JCT-VC website. Ex-1014, ¶¶35-37; Ex-1049.

282. The JCT-VC meetings were high-profile events that were widely known to those interested in video coding and were attended by individuals from around the United States and the world, including professors from universities and engineers for a wide range of technology companies. It was common for individuals and companies of the industry to monitor these standards meetings to

stay updated on changes and developments, as evidenced by several published papers citing Nakamura and WD4. *See, e.g.*, Ex-1024, Ex-1051, Ex-1025. Indeed, the paper "Non-fixed Quantization Considering Entropy Encoding in HEVC" by Gweon cites both Nakamura and WD4. Ex-1024, 000010.

283. Furthermore, the '833 patent admits that Nakamura and WD4 were prior art. First, the '833 patent references WD4 in its background. Ex-1001, 2:10-12 ("In some video codecs, such as High Efficiency Video Coding Working Draft 4…"). Second, Nakamura was cited as prior art against the '833 family of patents during prosecution of an European counterpart, EP2774375. *Supra* §II.B. The Patent Owner did not dispute that Nakamura was prior art and instead submitted Nakamura as prior art to the USPTO shortly before the '833 patent was allowed. *Supra* §II.B.

284. Therefore, for the reasons explained above, Nakamura and WD4 were publicly available before the provisional application for the '833 patent was filed on November 4, 2011.

285. Nakamura "presents simplifications of derivation process for merge mode and motion vector predictor (MVP)." Ex-1007, Abstract. The simplifications include "reduc[ing] the number of candidates in the spatial derivation process to reduce the number of times of comparison in the removal process." Ex-1007, Abstract. Additionally, Nakamura proposes an "improvement

of derivation method of the candidates for merge mode and motion vector predictor (MVP)" with a "unification of the location of spatial neighbors for merge mode and MVP." Ex-1007, §1. Nakamura is analogous art in the same field as the '833 patent (video encoding and decoding, *supra* §III). For example, Nakamura teaches "simplifications" and "improvement[s]" to video encoding and decoding processes for H.265. Ex-1007, Abstract, §1. Indeed, Nakamura provides results showing improvements in video encoding time and decoding time. Ex-1007, Tables 7-12.

286. WD4 contains various teachings related to the encoding and decoding of video streams according to the in-development HEVC video standard. For example, WD4 provides syntax for various video elements, such as prediction units, specifying "syntax element[s]… parsed from the bitstream[.]" Ex-1010, §7.1. WD4 is analogous art in the same field as the '833 patent (video encoding and decoding, *supra* §III). For example, WD4 describes aspects of video streams that were created by video encoders and decoded by video decoders, including the syntax of video bitstreams and video decoding processes. Ex-1010, §7.1, §8.4.

2. Motivation to Combine and Reasonable Expectation of Success

287. Nakamura is an HEVC proposal for "reduc[ing] the number of candidates in the spatial derivation process to reduce the number of … comparison[s] in the removal process." Ex-1007, Abstract. When applied to

WD4, Nakamura's teachings satisfy the challenged claims. A POSITA would have been motivated to apply Nakamura's teachings to the draft HEVC standard because that was its express purpose. Nakamura is a proposal presented to the JCT-VC, the standards body responsible for the HEVC standard, at the 6th JCT-VC Meeting on July 14-22, 2011. Ex-1007, Header. The meeting started with WD3 as the then-current version of the standard. At the meeting, various proposals were made including Nakamura, and the JCT-VC drafted WD4, the draft HEVC standard. Ex-1007, 000001; Ex-1010, 000001. Nakamura proposed "simplifications" and "improvement[s]" to the then-current working draft (WD3). Ex-1007, Abstract, Introduction, §6 (References). In view of these simplifications and improvements, WD4 "[i]ncorporated spatial merge candidate positions unification" from Nakamura. Ex-1010, Abstract. Therefore, the prior art provides express teaching, suggestion, and motivation to combine teachings from Nakamura and WD4.

288. Indeed, once the Nakamura proposal was made to the JCT-VC for inclusion into the draft HEVC standard, its teachings were known for HEVC. A POSITA would have found it obvious and been motivated to use Nakamura's teachings in the context of HEVC, including its working drafts. It would certainly not make sense for someone else to file for a patent many months later covering the application of Nakamura's teachings to the HEVC standard.

289. Nakamura was a proposal made at a JCT-VC meeting for the express purpose of applying its teachings to the HEVC standard, and WD4 was a draft version of the HEVC standard in the July 2011 timeframe. Nakamura teaches improvements to the HEVC standard and was intended to be used in the context of HEVC. Nakamura references aspects of the HEVC working drafts and uses HEVC terminology and concepts. Therefore, it would have been natural to apply Nakamura's teachings to draft HEVC standards; this would have applied a known technique to a known method ready for improvement to yield predictable results.

290. The combination of Nakamura and WD4 would have had predictable results. Nakamura teaches its "proposed technique is implemented into HM3.0 software", which was software implementation of the HEVC standard at that time. Ex-1007, §2.3.1. Furthermore, Nakamura provides results showing the actual application of its teachings to HEVC, including improvements from the combination. Ex-1007, §3 Therefore, Nakamura's teachings were readily applicable to HEVC and, indeed, were actually applied to HEVC. Thus, the combination of Nakamura and WD4 would have had predictable results. *See* Ex-1007, §3.

291. A POSITA would have been motivated to combine Nakamura and WD4 because it would have combined prior art elements according to known methods to yield predictable results. Nakamura teaches various improvements to

HEVC, including "simplifications of derivation process for merge mode and motion vector predictor (MVP)[,]… unification of the location of spatial neighbors for merge mode and MVP[,]" and "improvement of derivation method of the candidates for merge mode and motion vector predictor (MVP)." Ex-1007, Abstract, §1. These improvements involve, for example, "reduc[ing] the number of candidates in the spatial derivation process to reduce the number of times of comparison in the removal process." Ex-1007, Abstract. These teachings are applied to HEVC, which WD4 teaches. Ex-1007, Abstract, Introduction, §6 (References); Ex-1010, Abstract. Therefore, it would have been obvious to apply Nakamura's teachings to draft HEVC standards as a combination of prior art elements according to known methods, with predictable results.

292. In addition, a POSITA would have found motivation to combine Nakamura and WD4 because of the similarity of the references and the commonality of purpose. Nakamura and WD4 are both directed to video encoding and decoding, and are both meant for development of the HEVC standard. Both WD4 and Nakamura address motion vectors for block-based inter prediction. Ex-1010, §0.6; Ex-1007, Abstract. Nakamura teaches, for example, a "derivation process for merge mode and motion vector predictor (MVP)." Ex-1007, Abstract. WD4 also teaches the derivation process for motion vectors and adopts the techniques from Nakamura. Ex-1010, §8.4.2.1.1. Indeed, a POSITA would notice

striking similarities between Nakamura's "[d]erivation process[es]" and WD4's "[d]erivation process[es]." *Compare* Ex-1008 §§8.4.2.1.1-8.4.2.1.3 *with* Ex-1010, §§8.4.2.1.1-8.4.2.1.3. For example, Nakamura's "Derivation process for spatial merging candidates" and WD4's "Derivation process for spatial merging candidates" share striking similarities with a difference in labelling (e.g., Nakamura's A, B, C, D, E blocks and WD4's A_1, B_1, B_0, A_0, B_2 blocks) and Nakamura's inclusion of a figure being the only differences. *Compare* Ex-1008 §8.4.2.1.2 *with* Ex-1010, §8.4.2.1.2. Furthermore, Nakamura's "Derivation process of reference indices for temporal merging candidate" and WD4's "Derivation process of reference indices for temporal merging candidate" are identical. *Compare* Ex-1008 §8.4.2.1.3 *with* Ex-1010, §8.4.2.1.3.

293. Furthermore, a POSITA would have been motivated to combine WD4 and Nakamura because the teachings of WD4 and Nakamura are complementary. Nakamura provides detailed examples and explanations for deriving candidates for both merge mode and MVP mode. WD4 provides more context and details about how deriving candidates for merge mode and MVP mode work with the video encoding and decoding processes under the draft HEVC standard. *See generally* Ex-1010, §8. WD4 also explains the terminology and up-to-date background concepts of HEVC, especially on video encoding and decoding. *See generally, e.g.,* Ex-1010, §3. In understanding and implementing the teachings of Nakamura,

a POSITA would have been motivated to look to WD4, which provides additional technical context and implementation details.

294. A POSITA would have been motivated to combine Nakamura and WD4 because the combination would have yielded several advantages. For example, Nakamura and WD4 share the same goal of improving efficiency of video coding. WD4 teaches that "the need has arisen for an industry standard for compressed video representation with substantially increased coding efficiency…" Ex-1010, §0.1. Nakamura teaches that its proposal "reduce[s] the number of candidates in the spatial derivation process to reduce the number of times of comparison in the removal process." Ex-1007, Abstract. And Nakamura also provides actual results demonstrating the efficiency gains of its proposed techniques. Ex-1007, §3.

295. The combination of Nakamura and WD4 would not have changed the principle of operation for any of the teachings of the references relied upon in the combination. Nakamura was specifically proposed to be used with HEVC, which WD4 teaches, and that is precisely what the combination of Ground 2 is directed to. *See* Ex-1007, Abstract, §1, §6; Ex-1010, Abstract. The combination uses the teachings of Nakamura and WD4 in a conventional manner, as they were meant to be used. Moreover, both Nakamura and WD4 teach motion prediction for encoding and decoding blocks, with spatial motion vector prediction candidates.

Both Nakamura and WD4 operate on the same principles of block-based motion prediction. Ex-1023, 000004. Therefore, Nakamura's teachings would not have changed the principle of operation of WD4.

296. That Nakamura references an earlier version of HEVC (e.g., WD3) would not have dissuaded a POSITA from combining Nakamura with WD4. Indeed, the output of the 6th JCT-VC meeting, where Nakamura was presented, was WD4. A POSITA would have understood that Nakamura's teachings were applicable to HEVC, including the working drafts in that timeframe such as WD3 and WD4.

297. A POSITA would have had a reasonable expectation of success when combining Nakamura with WD4. As explained above, the combination applies teachings from each reference according to their known purposes, in a conventional manner as taught by Nakamura and WD4, without changing their principle of operation. Furthermore, Nakamura implemented the combination and reported measurements from actual results showing performance increases when Nakamura's teachings were applied to draft HEVC standards. Ex-1007, §2.3.1, §3; Ex-1009, 000015-000018. The combination does not modify Nakamura or WD4 in a way that would render either reference inoperative. Given that Nakamura had successfully made and reported the combination, a POSITA would have had a reasonable expectation of success in doing so. *See id.*

298. Given the education and experience of a POSITA (*supra* §III), a POSITA would have been more than capable of applying Nakamura's teachings to WD4, and vice versa, because it would simply have applied Nakamura's teachings, based on draft HEVC standards, to draft HEVC standards. Therefore, a POSITA would have had a reasonable expectation of success in applying Nakamura's teachings to draft HEVC standards, including WD4, which was after all the purpose of the Nakamura proposal for the HEVC standard.

299. Nakamura is a single proposal comprising multiple files. Each of those files is a component of the Nakamura proposal and therefore part of the reference. To the extent it is argued that Nakamura is a collection of documents, a POSITA would have been motivated to combine the teachings of the three Nakamura files relied on by this Declaration: Nakamura Main Document (Ex-1007), Nakamura WD Description (Ex-1008), and Nakamura Presentation (Ex-1009). First, these files were jointly presented to the JCT-VC in a single proposal and were meant to be read together to understand the Nakamura proposal. Second, the files were packaged together in a single zip file for download (Ex-1014 (Sze Declaration), ¶16), and therefore HEVC attendees would have understood that the files were meant to be read together as part of the Nakamura proposal. Third, the files teach related aspects of Nakamura's proposal and

therefore a POSITA would have understood from their context that their teachings were meant to be read and combined together. *See supra* §V.B.1.

300. A POSITA would also have had a reasonable expectation of success in combining the teachings of the files in the Nakamura proposal. As explained above, those files were drafted by the same author as part of the same proposal for the HEVC standard. Therefore, the teachings of Nakamura's files were meant to be read and used together, and a POSITA would have been capable of applying the teachings in the manner taught by the Nakamura files.

3. Independent Claim 9

[9pre] A method comprising:

301. To the extent that the preamble is limiting, Ground 2 teaches limitation [9pre], as explained below.

302. Nakamura teaches a **method**. For example, Nakamura teaches an "improvement of derivation method of the candidates for merge mode and motion vector predictor (MVP)." Ex-1007, §1, Abstract ("This contribution presents simplifications of derivation process for merge mode and motion vector predictor (MVP)."), §2.1, Fig. 1.

303. WD4 also teaches a method. For example, WD4 teaches a "[d]erivation process for spatial merging candidates." Ex-1010, §8.4.2.1.2.

304. Nakamura and WD4 further teach a method comprising the steps described below, with teachings for merge mode that satisfy the limitations as explained below. *Infra* §V.B.3[9pre]-[9g]. HEVC included two modes for predicting motion vector candidates: merge mode was optimal for areas of uniform motion, while MVP mode was more versatile but required more data. Generally, merge mode saved bits by utilizing the predicted motion vector without signaling difference vectors and other information used by MVP mode.

[9a]	**receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;**

305. Ground 2 teaches limitation [9a], as explained below.

306. Ground 2 teaches **receiving an encoded block of pixels including a prediction unit.** For example, WD4 teaches "allow[ing] motion video to be … transmitted and received over existing and future networks and distributed on existing and future broadcasting channels." Ex-1010, §0.2. Motion video is transmitted from an encoder to a decoder via a bitstream. Ex-1010, §3.11. The bitstream includes coded pictures which are divided into blocks of pixels. Ex-1010, §3.12, §3.21. The blocks of pixel include prediction units. Ex-1010, §7.3.7, §0.2, §3.11, §3.12, §3.34, §3.38. Therefore, the decoder receives an encoded block of pixels including a prediction unit via a bitstream from the encoder.

307. Ground 2 teaches **receiving … information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder.** For example, WD4 teaches a merge index (e.g., "merge_idx") that "specifies the merging candidate index of the merging candidate list[.]" Ex-1010, §7.4.7. Based on the merge_idx, one spatial "candidate at position merge_idx[xP][yP] in the merging candidate list mergeCandList" is assigned as the spatial candidate to use for prediction of the current block. Ex-1010, §8.4.2.1.1. WD4 teaches the merge_idx is received by the decoder from the encoder because WD4 teaches a "[p]rediction unit syntax" that describes what information is encoded for a prediction unit so it can be decoded. Ex-1010, §7.3.7 (red square highlights added; other markings in original):

prediction_unit(x0, y0~~, log2PUWidth, log2PUHeight, PartIdx~~) {	**Descriptor**
if(skip_flag[x0][y0]) {	
merge_idx[x0 **][** y0 **]**	ue(v) | ae(v)
} else if(PredMode == MODE_INTRA) {	
if(PartMode == PART_2Nx2N && log2CUSize~~PUWidth~~ >= Log2MinIPCMCUSize)	
pcm_flag	u(1) | ae(v)
if(pcm_flag) {	
while (!byte_aligned( ))	
pcm_alignment_zero_bit	u(v)
for(i = 0; i < 1 << (log2CUSize << 1); i++)	
pcm_sample_luma[i **]**	u(v)
for(i = 0; i < (1 << (log2CUSize << 1)) >> 1; i++)	
pcm_sample_chroma[i **]**	u(v)
} else {	
prev_intra_luma_pred_flag[x0 **][** y0 **]**	u(1) | ae(v)
if(prev_intra_luma_pred_flag[x0][y0])	
if(NumMPMCand > 1)	
mpm_idx[x0 **][** y0 **]**	u(1) | ae(v)
else	
rem_intra_luma_pred_mode[x0 **][** y0 **]**	ce(v) | ae(v)
if(IntraPredMode[x0][y0] == 2)	
planar_flag_luma[x0 **][** y0 **]**	u(1) | ae(v)

intra_chroma_pred_mode[x0][y0]	ue(v) | ae(v)
SignaledAsChromaDC = (chroma_pred_from_luma_enabled_flag ? intra_chroma_pred_mode[x0][y0] == 3 : intra_chroma_pred_mode[x0][y0] == 2)	
if(IntraPredMode[x0][y0] != 2 && IntraPredMode[x0][y0]!=34 && SignaledAsChromaDC)	
planar_flag_chroma[x0][y0]	u(1) | ae(v)
}	
} else { /* MODE_INTER */	
if(entropy_coding_mode_flag || PartMode != PART_2Nx2N)	
merge_flag[x0][y0]	u(1) | ae(v)
if(merge_flag[x0][y0]) {	
merge_idx[x0][y0]	ue(v) | ae(v)
} else {	
if(slice_type == B) {	
if(!entropy_coding_mode_flag) {	
combined_inter_pred_ref_idx	ue(v)
if(combined_inter_pred_ref_idx == MaxPredRef)	
inter_pred_flag[x0][y0]	ue(v)
} else	
inter_pred_flag[x0][y0]	ue(v) | ae(v)
}	
if(inter_pred_flag[x0][y0] == Pred_LC) {	
if(num_ref_idx_lc_active_minus1 > 0) {	

The prediction unit syntax describes information encoded for a prediction unit. For example, for a "prediction_unit(x0, y0)" or a prediction unit located at x0, y0, a "skip_flag[x0][y0] equal to 1 specifies that for the current coding unit… no more syntax elements except the motion vector predictor indices[,]" which is merge_idx, "are parsed[.]" Ex-1010, §7.4.6. As another example, "merge_flag[x0][y0] specifies whether the inter prediction parameters for the current prediction unit are inferred from a neighbouring inter-predicted partition[,]" which is merge mode. Ex-1010, §7.4.7. If merge_flag is set equal to 1, then "merge_idx[x0][y0] specifies the merging candidate index of the merging candidate list[.]" Ex-1010,

7.4.7. The prediction unit is encoded into the bitstream that is received by a decoder. Ex-1010, §7.3.7, §0.2, §3.11, §3.12, §3.34, §3.38.

308. A POSITA would have known, and it would have been obvious, that the merging candidate list, mergeCandList, is constructed by the encoder. As seen from above, merge_idx is generated in the encoding process and is stored as part of an encoded video, since it is part of information encoded for a prediction unit. The encoder generates the list of spatial motion vector prediction candidates so that the merge_idx points to a specific merging candidate used for predicting the motion vector for the current block. Ex-1008, §8.4.2.1.1; Ex-1010, §7.3.7, §7.4.7, §8.4.2.1.1.

309. As merge_idx identifies the merging candidate index of the merging candidate list, it is for use in decoding the encoded block by the decoder. WD4 teaches that, in the derivation process for luma motion vectors, the decoder uses merge_idx to select a merging candidate as the spatial candidate assigned as the motion vector predictor for the current prediction unit. WD4 teaches that this is achieved by setting the motion vector information (e.g., mvLX), reference index (e.g., refIdxLX), and prediction list utilization flags (e.g., predFlagLX) to those of the spatial candidate.:

> The following assignments are made with N being the candidate at position merge_idx[xP][yP] in the merging candidate list

mergeCandList (N = mergeCandList[merge_idx[xP][yP]]) and X being replaced by 0 or 1:

mvLX[0] = mvLXN[0] (8 88)

mvLX[1] = mvLXN[1] (8 89)

refIdxLX = refIdxLXN (8 90)

predFlagLX = predFlagLXN (8 91)

Ex-1010, §8.4.2.1.1.

310. Therefore, WD4 teaches using merge_idx to select a merging candidate from the merging candidate list, which is constructed by the encoder. Accordingly, merge_idx is information received by WD4's decoder to identify a respective spatial motion vector prediction candidate from the merge list constructed by an encoder. WD4 teaches using merge_idx to select spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels. Nakamura and WD4 teach, or at least suggest, that the decoder receives merge_idx from the encoder because it is used by the encoder to create the video bitstream and saved as a syntax element into the bitstream, which is then received by the decoder. Ex-1010, §3.11, §3.12, §3.34, §3.38, §8.4.2.1.1. The encoder and decoder independently construct identical merge lists (mergeCandList) and use merge_idx to identify which candidate in the list was used for a particular PU. Ex-1008, §8.4.2.1.1; Ex-1010, §7.3.7, §7.4.7, §8.4.2.1.1.

311. Nakamura confirms that its merge mode teachings are used in the encoding and decoding processes by providing experimental results showing improvements at both the encoder and the decoder based on its teachings. Ex-1007, Tables 5-8, 11-12; Ex-1009, 000016, 000018; *see also* §V.B.2 (explaining that the teachings of motion vector derivation are similar in Nakamura and WD4). At the very least, a POSITA would have found it obvious that the decoder would receive the merge_idx with an encoded video so that, when the decoder receives the video transmitted from the encoder, the decoder can use the merge_idx to decode the video.

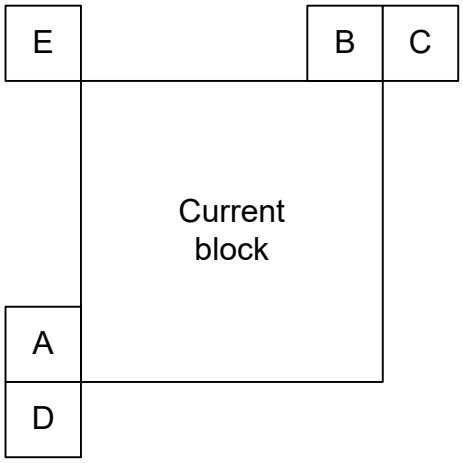
312. Based on the foregoing explanations, Ground 2 teaches limitation [9a].

[9b]	**determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

313. Ground 2 teaches limitation [9b], as explained below.

314. Ground 2 teaches **determining a set of spatial motion vector prediction candidates for the encoded block of pixels.** For example, Nakamura teaches a process for identifying two spatial motion vector prediction candidates (S_0 and S_1) from a set of motion vector candidates from "the position[s] of the spatial neighbors A, B, C, D and E relative to the current prediction unit" which "can be used for… candidates" for the current block. Ex-1007, §2.2.1, Abstract,

§1, Tables 2, 4, Figure 1; Ex-1009, 000008, 000010, 000014. The set of motion vector candidates are evaluated, in sequence, from the five neighboring blocks (A, B, C, D, E). *E.g.*, Ex-1007, 000004-6 (Tables 2-4); Ex-1009, 000008, 000010, 000012-14. The motion vector candidates are spatial motion vector prediction candidates because each is a candidate motion vector obtained from one or more previously-encoded blocks in the current frame (A, B, C, D, and E). *E.g.*, Ex-1007, §2.2.1; *Supra* §IV.A. As illustrated in Figure 2(b), the spatial neighbors are blocks A, B, C, D, and E, which neighbor the current block of pixels. Ex-1007, Figure 2(b):



(b) Proposed technique

315. Since blocks are processed in raster scan order from top left to bottom right, blocks A-E represent the potential neighboring blocks that have already been encoded. Blocks below and to the right of the current block will not have been encoded yet and are not available as sources for spatial motion vector prediction

candidates. *See* Ex-1010, §3.83 (explaining "raster scan"), §3.101. Furthermore, Nakamura and WD4's teachings "enable a high compression capability for a desired image or video quality" using "block-based inter prediction[,]" which teaches the blocks are blocks of pixels because images and videos are comprised of pixels. Ex-1010, §0.6.

316. WD4 teaches that a "decoding process specified in this Recommendation | International Standard that reads a bitstream and derives decoded pictures from it" and an "encoding process… that produces a bitstream conforming to this Recommendation | International Standard." Ex-1010, §3.34, §3.38. Therefore, the teachings of Nakamura and WD4 for encoding a video and decoding the video are applicable to both processes so that a bitstream encoded based on these teachings would be correctly decoded. Thus, the teachings of Nakamura and WD4 apply to an encoded block of pixels.

317. Ground 2 teaches **the spatial motion vector prediction candidates being provided with motion information.** For example, each spatial motion vector prediction candidate comprises "availability flags… reference indices… prediction list utilization flags…" and "*motion vectors*" for prediction. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. This information is used as inputs in a "decoding process for prediction units" to derive the "samples" (e.g., pixels) for a block of

pixels. Ex-1010, §8.4.2, §3. Thus, each spatial motion vector prediction candidate is provided with motion information.

318. Based on the foregoing explanations, Ground 2 teaches limitation [9b].

[9c]	**selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

319. Ground 2 teaches limitation [9c], as explained below.

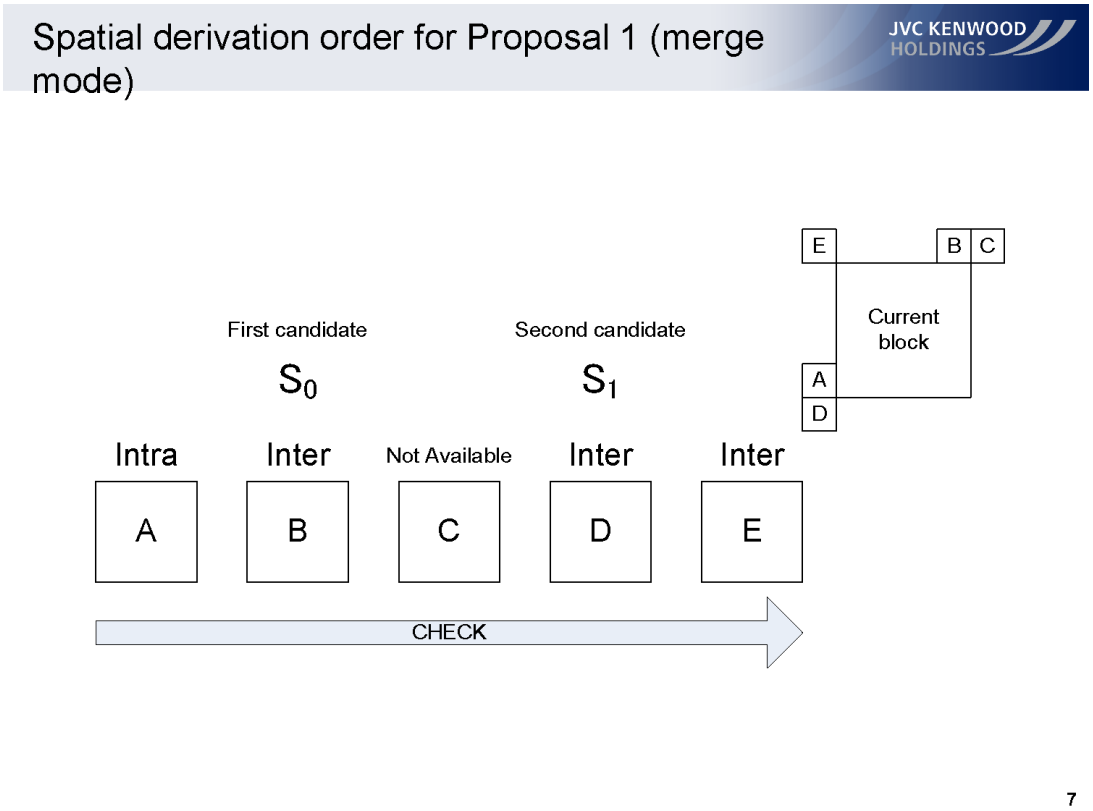
320. Ground 2 teaches **selecting a first spatial motion vector prediction candidate** (e.g., for S_1) **from the set of spatial motion vector prediction candidates**. For example, to identify two spatial motion vector prediction candidates (S_0 and S_1) for the merging candidates list, Nakamura steps through each of the spatially-neighboring blocks (A, B, C, D, E) in sequence to select S_0 from one of the blocks and S_1 from one of the remaining spatial neighbors. Ex-1007, Tables 2, 4 ("Spatial derivation order"); Ex-1009, 000008. In particular, as explained further below, the selection of a candidate from a spatial neighbor for S_1 satisfies the claimed step for selecting a first spatial motion vector prediction candidate. It is irrelevant whether Nakamura applies its own "first" and "second" labels to describe the ordering of its spatial derivation process, or whether any other candidates come before S_1, because claim 1 merely recites "*a first* spatial

motion vector prediction candidate" to distinguish that candidate from other spatial motion vector prediction candidates described in the claims, not as an absolute temporal requirement for that candidate. Therefore, it is my understanding that claim 1 does not impose a requirement for selecting the very first spatial motion vector prediction candidate for a video, frame, or particular block. Instead, this language simply requires selecting *a* spatial motion vector prediction candidate, which is then designated the "first" spatial motion vector prediction candidate for the remainder of the claim. Here, the selected first spatial motion vector prediction candidate (as recited in claim 1) is Nakamura's S_1 for purposes of analyzing claim 9.

321. Ground 2 teaches **selecting a first… candidate… as a potential spatial motion vector prediction candidate to be included in a merge list**. For example, Nakamura teaches a "merging candidate list is constructed of two spatial merging candidates" (S_0 and S_1) selected from the five spatial neighbor candidates "relative to the current prediction unit[.]" Ex-1007, §2.2.1, Table 2, §2.2.2, Table 3; Ex-1008, §8.4.2.1.1 ("The merging candidate list, mergeCandList…").

322. To select S_0 and S_1, Nakamura teaches a "[s]patial derivation order" in which candidates from the five spatially neighboring blocks are evaluated in sequence. Ex-1007, Table 2-4; Ex-1009, 000008, 000010, 000012-14. Nakamura

illustrates an example of the "[s]patial derivation order" which starts from A and progresses to E. Ex-1009, 000008:



323. The above example illustrates the order in which the spatial neighbors are checked. Ex-1009, 000008. Following the spatial derivation order of A through E, Nakamura begins with block A and decides whether its candidate should be selected for S_0. Since A is an intra block and does not have a motion vector, Nakamura proceeds to the next block B, whose spatial motion vector prediction candidate is selected as candidate S_0 for the merging candidate list because B is an inter block with motion vector prediction information. Nakamura proceeds in this manner, stepping through the spatial derivation order to select S_1,

until two spatial candidates (S_0 and S_1) are selected for the merging candidate list. Ex-1009, 000008; Ex-1007, §2.2.1, Table 2. In this example, block C is skipped because it is not available. The spatial motion vector prediction candidate from Block D is then evaluated and selected for S_1. Block E is not checked because two spatial candidates have been found before the spatial derivation order reaches block E. Ex-1009, 000008. Therefore, Nakamura teaches selecting a first spatial motion vector prediction candidate (e.g., for S_1).

324. The merging candidate list is **a merge list for the prediction unit.** Nakamura teaches "[t]he merging candidate list, mergeCandList," (Ex-1008, §8.4.2.1.1; Ex-1010, §8.4.2.1.1) comprises "spatial merging candidates" found from "the position[s] of the spatial neighbors… relative to the current prediction unit" of the current block. Ex-1007, §2.2.1, Tables 2, 4, Figure 2(b). One of the spatial candidates in the merging candidate list is assigned as the motion vector predictor for the current prediction unit. Ex-1008, §8.4.2.1.1 ("The following assignments are made with N being the candidate at position mergeIdx in the merging candidate list mergeCandList (N = mergeCandList[mergeIdx])…"); Ex-1010, §7.4.7 (explaining that merge_idx "specifies the merging candidate index of the merging candidate list" for a prediction unit), §8.4.2.1.1, 000049, 000174 (showing "merge_idx" is associated with a prediction unit). Therefore, the

merging candidate list is a merge list for the PU corresponding to the block currently being processed. *See, e.g.,* Ex-1008, §8.4.2.1.1.

325. Nakamura teaches a "merging candidate list," which is referred to in WD4 as "mergeCandList." A POSITA reading Nakamura and WD4 would have understood this because Nakamura was a proposal for HEVC, and "mergeCandList" is an abbreviation for Merge Candidate List following a common computer programming style. Ex-1008, §8.4.2.1.1; Ex-1010, §8.4.2.1.1 ("The merging candidate list, mergeCandList…").

326. **Where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates.** Nakamura teaches that the "merging candidate list is constructed of" spatial candidates (Ex-1007, §2.2.1) and includes information of those candidates in the list, including "availability flags… reference indices… prediction list utilization flags…" and "motion vectors[.]" Ex-1008, §8.4.2.1.2. As part of the process for building the merging candidate list, Nakamura teaches comparing the two spatial candidates (S_0 and S_1) to "[r]emove candidates with the same motion information[.]" Ex-1007, Figure 1, Table 4, Table 2; *infra* §V.A.3[9e]. Therefore, the merging candidate list is constructed based on the motion information of the two spatial candidates.

327. Based on the foregoing explanations, Ground 2 teaches limitation [9c].

[9d]	**determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

328. Ground 2 teaches limitation [9d], as explained below.

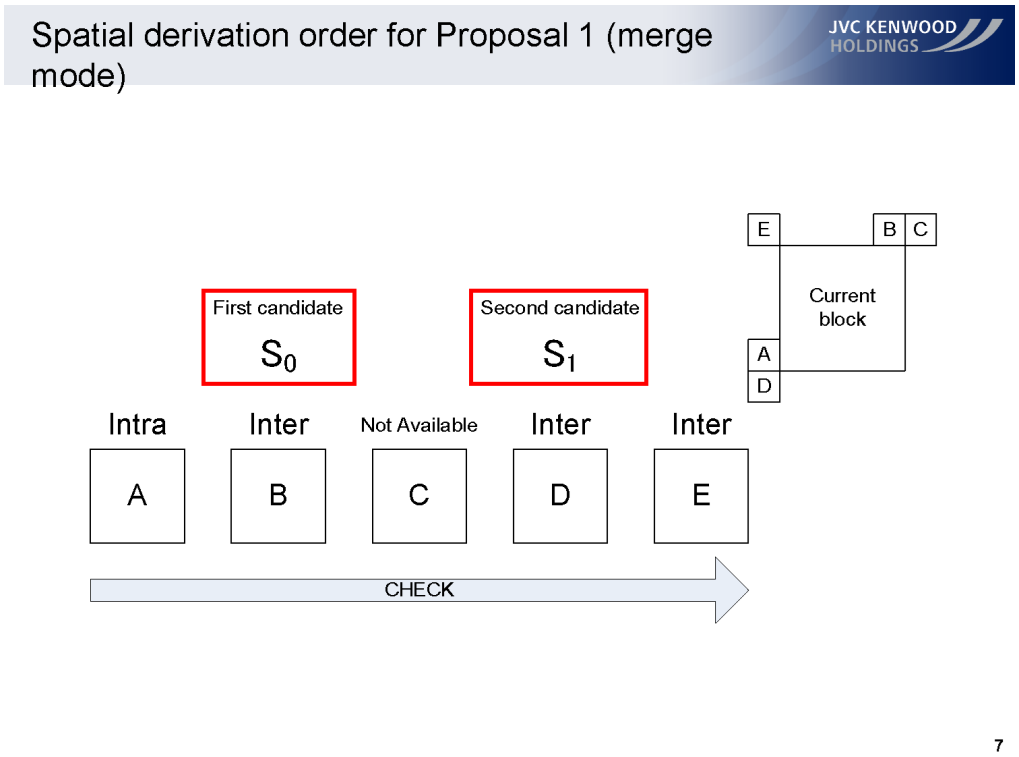
329. Ground 2 teaches **determining a subset of spatial motion vector prediction candidates.** For example, Nakamura derives two candidates out of the candidates from five spatially-neighboring blocks. For example, Nakamura teaches that "[t]wo spatial merging candidates are derived in the spatial derivation process[]" from five "spatial neighbors A, B, C, D, and E relative to the current prediction unit[.]" Ex-1007, §2.2.1, Tables 1, 4; Ex-1009, 000010, 000014; Ex-1007, Table 2 ("2 in 5 [positions]", "S_0, S_1"); Ex-1009, 000008; *supra* §V.D.

330. After determining the two spatial merging candidates, Nakamura teaches a removal process that compares S_0 with S_1 to check for redundancy when determining whether to include or exclude S_1. Ex-1007, Tables 2, 4 ("S_0 vs S_1"); Ex-1008, §8.4.2.1.1; Ex-1009, 000010, 13; Ex-1010, §8.4.2.1.1. Therefore, Nakamura determines a subset of spatial motion vector prediction candidates (e.g., S_0).

Table 2 Comparison between HM3.0 and proposed technique for merge mode

	HM3.0	Proposal 1
The number of spatial candidates	4 in 4 [positions]	2 in 5 [positions]
Spatial derivation order	A, B, C, D	A, B, C, D, E
The number of times of comparison of redundant candidates in the spatial derivation process	0 [time]	0 [time]
The number of temporal candidates	1	1
Merging candidate list order	A, B, Col, C, D	S_0, S_1, Col
The number of times of comparison in the removal process	10 [times] (A vs B, Col, C, D, B vs Col, C, D, Col vs C, D, and C vs D)	3 [times] (S_0 vs S_1, S_0 vs Col, and S_1 vs Col)

331. Additionally, as explained for [9b], Nakamura proceeds sequentially to select S_0 and then S_1. When S_1 is being selected, the merge list includes a subset of one spatial motion vector prediction candidate (S_0). *E.g.*, Ex-1009, 000008, 000010; Ex-1007, Table 2; *supra* §V.B.3[9b]. This is a subset of spatial motion vector prediction candidates (e.g., S_0). Ex-1009, 000008; *see also* Ex-1007, §2.2.1, Ex-1009, 000010:



332. The determination is **based on the location of the block associated with the first spatial motion vector prediction candidate.** Here, "the" block recited in [9d] has an antecedent basis in [9a]: "*an encoded block of pixels*[.]" Therefore, the block associated with the first spatial motion vector prediction candidate is the current block for which motion vector prediction candidates are being analyzed. *Supra* §IV.C.

333. Nakamura determines its subset based on the location of the current block because the subset is determined from a set of "spatial neighbors A, B, C, D and E relative to the current prediction unit" for the current block. Ex-1007, §2.2.1, Fig. 2(b). The spatial neighbors A, B, C, D and E are all determined from their position relative to the current block—they are the neighboring blocks above

and to the left of the current block. Ex-1007, §2.2.1; Ex-1009, 000008. S_0 and S_1 are both selected from these spatial neighbors; therefore, S_0 and S_1 are also determined based on the location of the current block by analyzing candidates from spatial neighbors of the current block. Ex-1007, §2.2.1, Fig. 2(b).

334. Additionally, the subset comprising candidate S_0 is determined based on the location of the current block: it is the first available spatial motion vector prediction candidate from the neighboring blocks of the current block, following Nakamura's spatial derivation order which is defined based on the relative position of blocks around the current block. Ex-1009, 000008, 000010; Ex-1007, §2.2.1, Fig. 2; Ex-1008, §8.4.2.1.2, Fig. 8-3.

335. In the HEVC context, each PU corresponds with a block of pixels. Ex-1010, §6.3; *see also* Ex-1005, ¶33. The merge mode process iteratively steps through blocks in a frame, and in each step, evaluates spatial motion vector prediction candidates for the current block. Ex-1007, §2.2.1; Ex-1008, §8.4.2.1.1 (constructing mergeCandList in a specified order); Ex-1010, §8.4.2.1.1.

336. Based on the foregoing explanations, Ground 2 teaches limitation [9d].

[9e]	**comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the [sub]set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric (incorporating certificate of correction);**

337. Ground 2 teaches limitation [9e], as explained below.

338. Ground 2 teaches **comparing motion information of the first spatial motion vector prediction candidate** (e.g., for S_1) **with motion information of another spatial motion vector prediction candidate** (e.g., for S_0) **of the [sub]set of spatial motion vector prediction candidates … wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric**. For example, WD4 removed duplicate candidates from the merge list. Ex-1010, §8.4.2.1.1. Consistent with those teachings, Nakamura teaches comparing the two spatial candidates (S_0 and S_1) to "[r]emove candidates with the same motion information[.]" Ex-1007, Fig. 1, Table 4, Table 2:

Table 2 Comparison between HM3.0 and proposed technique for merge mode

	HM3.0	Proposal 1
The number of spatial candidates	4 in 4 [positions]	2 in 5 [positions]
Spatial derivation order	A, B, C, D	A, B, C, D, E
The number of times of comparison of redundant candidates in the spatial derivation process	0 [time]	0 [time]
The number of temporal candidates	1	1
Merging candidate list order	A, B, Col, C, D	S_0, S_1, Col
The number of times of comparison in the removal process	10 [times] (A vs B, Col, C, D, B vs Col, C, D, Col vs C, D, and C vs D)	3 [times] (S_0 vs S_1, S_0 vs Col, and S_1 vs Col)

339. As illustrated in Table 2, above, the overall removal process for WD3 required 10 comparisons because the merge list included 5 candidates (4 spatial candidates A-D plus a temporal candidate "Col"), and each pair from the set of 5 had to be compared to identify all redundant candidates. Since there were 10 possible pairings for the five candidates, 10 comparisons were needed for WD3 to determine if there were duplicates. Ex-1007, Fig. 1, Table 4, Table 2.

340. Nakamura reduced the number of spatial candidates to 2, meaning the merge list included 3 candidates total: two spatial candidates (S_0 and S_1) plus a temporal candidate "Col." Ex-1007, Fig. 1, Table 4, Table 2; Ex-1009, 000010. Since there were only 3 candidates, the removal process only required three comparisons: one comparison of the two spatial candidates (S_0 and S_1), one

comparison of the temporal candidate with S_0, and one comparison of the temporal candidate with S_1. Ex-1007, Table 2.

341. The comparison between the spatial candidates (S_0 and S_1) involves comparing their motion information, which includes "motion vectors" and "reference indices[.]" Ex-1010, §8.4.2.1.1 ("When merging candidates have the [same] motion vectors and the same reference indices, the merging candidates are removed from the list except the merging candidate which has the smallest order in the mergeCandList."). The comparison involves an equivalence check in order to "[r]emove candidates with the same motion information[.]" Ex-1007, Fig. 1, Table 4, Table 2. If S_0 and S_1 are equivalent and have the same motion information, then one is removed.

342. In addition, while not required by the Challenged Claims, a POSITA would have found it obvious to apply Nakamura's teachings, for removing duplicates, when spatial motion vector prediction candidates (e.g., S_0 and S_1) are identified and considered for potential inclusion in the merge list. This would have been obvious because it is a straightforward way to prevent redundant candidates in the merge list, as Nakamura and WD4 teach. Ex-1007, Fig. 1. Moreover, Nakamura teaches comparing S_0 with S_1 and then comparing the temporal candidate with the spatial candidates (Ex-1007, Table2, Table 4; Ex-1009, 000010, 000013; Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2); this is consistent with an

implementation that compares a potential candidate with those already in the merge list when the potential candidate is being considered. For example, when evaluating S_1, it is compared with the only candidate in the merge list S_0; when evaluating the temporal candidate, it is compared with the two spatial candidates in the merge list, resulting in a maximum of the three comparisons, as Nakamura teaches. Ex-1007, Fig. 1, Table 4, Table 2.

343. Ground 2 teaches **comparing… without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates.** For example, Nakamura makes the above-described comparison (between S_0 and S_1) without making a comparison of each pair from the set of spatial candidates (from blocks A, B, C, D, and E). Ex-1007, Tables 1-2, 4. Indeed, that is the purpose of Nakamura's proposal, to reduce the number of comparisons by avoiding a comparison of each pair of candidates from blocks A, B, C, D, and E. *E.g.*, Ex-1007, Fig. 1, Tables 2, 4. Nakamura teaches that reducing "[t]he number of candidates… in the spatial derivation process… reduce[s] the number of times of comparison in the removal process." Ex-1007, §2.2.1, Table 1. Thus, the application of these teachings from Nakamura improves coding efficiency by reducing the number of comparisons needed for removing duplicates.

344. Based on the foregoing explanations, Ground 2 teaches limitation [9e].

[9f]	**if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and**

345. Ground 2 teaches limitation [9f], as explained below.

346. Nakamura teaches "[r]emov[ing] candidates with the same motion information" in the removal process. Ex-1007, Fig. 1(a). Nakamura compares motion information for the first spatial motion vector prediction candidate (S_1) with another spatial motion vector prediction candidate (S_0). Ex-1007, Fig. 1, Tables 2, 4; Ex-1010, §8.4.2.1.1; *supra* §V.B.3[9c]. Nakamura determines whether to exclude S_1 in the merge list based on this comparison: if the motion information is different, then both S_0 and S_1 are included in the merging candidate list. Ex-1007, Fig. 1; Ex-1010, §8.4.2.1.1. Conversely, if the motion information is the same, then S_1 corresponds with S_0, and Nakamura excludes S_1 in the merging candidate list based on the comparison while S_0 is included because it has the smaller order (it is earlier in the spatial derivation order): "candidates are removed from the list except the merging candidate which has the smallest order in the mergeCandList[.]" Ex-1010, §8.4.2.1.1.

347. Based on the foregoing explanations, Ground 2 teaches limitation [9f].

[9g]	**selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

348. Ground 2 teaches limitation [9g], as explained below.

349. Ground 2 teaches **selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels**. For example, WD4 teaches a merge index (e.g., "merge_idx") that "specifies the merging candidate index of the merging candidate list[.]" Ex-1010, §7.4.7. Based on the merge_idx, one spatial "candidate at position merge_idx[xP][yP] in the merging candidate list mergeCandList" is assigned as the spatial candidate to use for prediction of the current block. Ex-1010, §8.4.2.1.1. As I also explained above, the spatial merging candidate contains motion information such as the motion vector and reference index and is used for the prediction of a block. *See supra* §V.B.3[9a]. Therefore, WD4 teaches using merge_idx to select the spatial candidate from the merge list for use in decoding the encoded block of pixels.

350. Ground 2 teaches **wherein the spatial motion vector prediction candidate is selected from the merge list using information that was received identifying a respective spatial motion vector prediction candidate**. Here, the "information that was received identifying a respective spatial motion vector prediction candidate" references the "information identifying a respective motion

vector prediction candidate from a merge list constructed by an encoder" in limitation [9a] and is taught by Ground 2 for the reasons explained for limitation [9a]. *Supra* §V.B.3[9a]. WD4 teaches that merge_idx, the information that identifies the merging candidate index of the merging candidate list, is used to select a merging candidate as the spatial candidate assigned as the motion vector predictor for the current prediction unit. WD4 teaches that this is achieved by setting the motion vector information (e.g., mvLX), reference index (e.g., refIdxLX), and prediction list utilization flags (e.g., predFlagLX) to those of the spatial candidate:

> The following assignments are made with N being the candidate at position merge_idx[xP][yP] in the merging candidate list mergeCandList (N = mergeCandList[merge_idx[xP][yP]]) and X being replaced by 0 or 1:
>
> mvLX[0] = mvLXN[0] (8 88)
>
> mvLX[1] = mvLXN[1] (8 89)
>
> refIdxLX = refIdxLXN (8 90)
>
> predFlagLX = predFlagLXN (8 91)

Ex-1010, §8.4.2.1.1.

351. Therefore, WD4 teaches using merge_idx to select a merging candidate from the merging candidate list. The encoder and decoder independently construct identical merge lists (mergeCandList) and use merge_idx to identify

which candidate in the list was used for a particular PU. Ex-1008, §8.4.2.1.1; Ex-1010, §7.3.7, §7.4.7, §8.4.2.1.1.

352. Based on the foregoing explanations, Ground 2 teaches limitation [9g].

4. Dependent Claim 10

10.	**The method according to claim 9 comprising comparing motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.**

353. Ground 2 teaches claim 10, as explained below.

354. Ground 2 teaches **comparing motion information of the potential spatial motion vector prediction candidate** (e.g., S_1) **with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates** (e.g., S_0). For example, Nakamura teaches comparing motion information of two spatial candidates (S_0 and S_1) as part of a removal process that "[r]emove[s] candidates with the same motion information[.]" Ex-1007, Figure 1, Tables 2, 4; *supra* §V.B.3[9c]. In the removal process, "[t]he number of times of comparison" is 3, with only one of the comparisons being between the two spatial candidates. Ex-1007, Tables 4, 2:

Table 2 Comparison between HM3.0 and proposed technique for merge mode

	HM3.0	Proposal 1
The number of spatial candidates	4 in 4 [positions]	2 in 5 [positions]
Spatial derivation order	A, B, C, D	A, B, C, D, E
The number of times of comparison of redundant candidates in the spatial derivation process	0 [time]	0 [time]
The number of temporal candidates	1	1
Merging candidate list order	A, B, Col, C, D	S_0, S_1, Col
The number of times of comparison in the removal process	10 [times] (A vs B, Col, C, D, B vs Col, C, D, Col vs C, D, and C vs D)	3 [times] (S_0 vs S_1, S_0 vs Col, and S_1 vs Col)

355. Potential spatial motion vector prediction candidate S_1 is only compared to one other spatial motion vector prediction candidate (S_0). Since "[t]he merging candidate list is constructed of two spatial merging candidates[,]" S_1 can be compared to at most one other spatial motion vector prediction candidate during the removal process. Ex-1007, §2.2.1. Table 2 confirms this: "comparison[s] in the removal process" are **S_0 vs S_1**, S_0 vs Col, and S_1 vs Col, with S_0 and S_1 being the "spatial candidate[s] found in the spatial derivation process" and Col being a co-located temporal candidate. Ex-1007, Table 2, §2.2.1; *see also* Ex-1008, §8.4.2.1.1. Since Col is a temporal candidate, it is not a spatial motion vector prediction candidate and is not part of the set of spatial motion vector prediction candidates.

356. Therefore, Nakamura teaches comparing motion information of the potential spatial motion vector prediction candidate (e.g., S_1) with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates (e.g., S_0).

357. Based on the foregoing explanations, Ground 2 teaches claim 10.

5. Dependent Claim 11

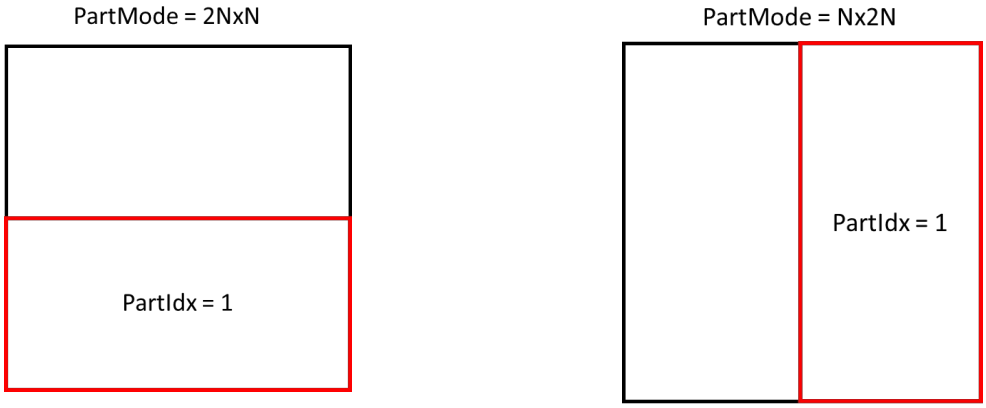
11.	**The method according to claim 9 comprising examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.**

358. Ground 2 teaches claim 11.

359. Ground 2 teaches **examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit**. For example, Nakamura and WD4 teach excluding a spatial candidate, by setting it as unavailable with "availableFlagN is set equal to 0[,]" if "one of the following conditions is true[:]… PartMode of the current prediction unit is PART_2NxN and PartIdx is equal to 1…" or "PartMode of the current prediction unit is PART Nx2N and PartIdx is equal to 1[.]" Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. In these passages, Nakamura examines whether the block of pixels is divided into a first

and second PU ("PartMode of the current prediction unit" is PART_2NxN or PART Nx2N) and whether the current PUis the second PU ("PartIdx is equal to 1"). If so, Nakamura excludes the potential spatial motion vector prediction candidate from block B (for vertically-divided blocks) or block A (for horizontally-divided blocks) from the merge list. This is further explained below.

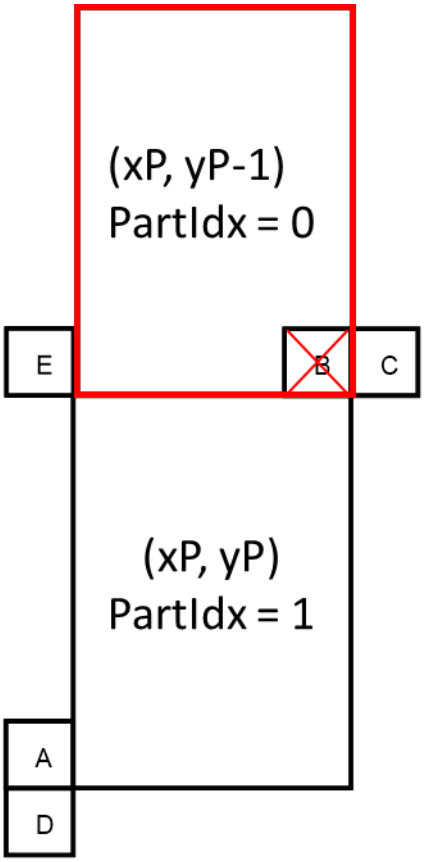
360. For HEVC, partition mode PART_2NxN refers to a block that is vertically divided into two equal PUs; PART_Nx2N refers to a block horizontally divided into two equal PUs. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2; *supra* §I.C (explaining partition notation). Each PU has a partition ID. The first PU has PartIdx = 0, while the second PU has PartIdx = 1. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Nakamura checks if "PartIdx is equal to 1" meaning it checks if the PU is the second PU of the divided block. I have illustrated this below:



361. If the block is divided into a first and second PUs, then Nakamura excludes the potential spatial motion vector prediction candidate from the merge list if the current PU is the second PU. Ex-1008, §8.4.2.1.2. This is explained

below for the vertically-divided block (2NxN) and then the horizontally-divided block (Nx2N).

362. For the 2NxN (vertically-divided) scenario, when deciding the availability of neighboring blocks A-E, Nakamura sets the spatial candidate for block B and any spatial candidate that has "identical motion parameters[]" as block B to unavailable, meaning it is excluded from the merge list. Ex-1008, §8.4.2.1.2. Nakamura iterates through neighboring blocks, setting variable "N" to A, B, C, D or E in sequence. *Id.* ("the derivation of availableFlagN, with N being A, B, C, D or E"). For each block, including block B, Nakamura evaluates whether the block has "identical motion parameters" with block B. *Id.* ("PartMode of the current prediction unit is PART_2NxN and PartIdx is equal to 1 and *the prediction units covering luma location (xP, yP–1) (PartIdx=0)* and luma location (xN,yN) (Cand. N) have *identical motion parameters*[.]"); Ex-1010, §8.4.2.1.2. Here, the coordinates "(xP, yP–1)" refers to the block above the current block because it has the same x coordinate but is 1 row up. The y coordinate has 1 subtracted from it, which results in the previous block in the vertical (y) direction. Since block B is the block above the current block, block B is a PU "covering… (xP, yP-1)." *Id.* I have illustrated this below, with a red box showing the block that Nakamura would exclude:

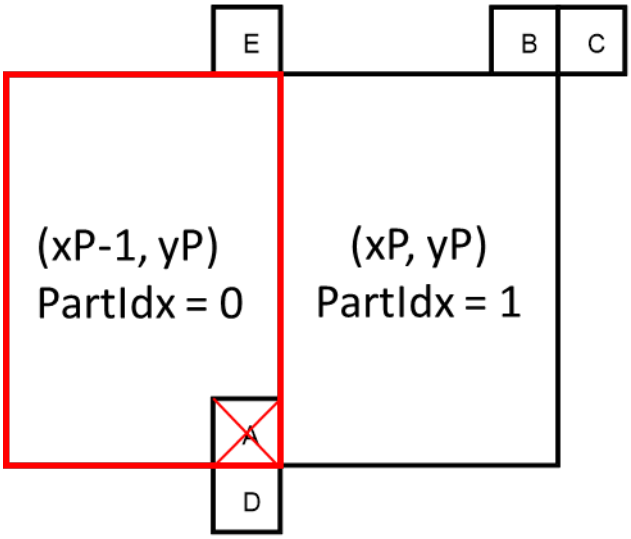


363. Block B has "PartIdx=0" because this is a vertically-divided block (part mode 2NxN), and block B is above the current block, which has PartIdx=1. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Therefore, when Nakamura iterates to block B, then N is set to B (Ex-1008, §8.4.2.1.2 ("the derivation of availableFlagN, with N being A, B, C, D or E")), and Nakamura excludes block B's potential spatial motion vector prediction candidate from the merge list because, by the law of identity, block B has "identical motion parameters" with itself. *Id.*

364. By setting availableFlagB "equal to 0," block B is not available for, and therefore excluded from, the merging candidate list. Ex-1008, §8.4.2.1.2

("The merging candidate list, mergeCandList, is constructed of… B, if availableFlagB is equal to 1[.]"); Ex-1010, §8.4.2.1.2.

365. For the Nx2N (horizontally-divided) scenario, when deciding the availability of neighboring blocks A-E, Nakamura sets the spatial candidate for block A and any spatial candidate that has "identical motion parameters[]" as block A to unavailable, meaning it is excluded from the merge list. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Nakamura iterates through neighboring blocks, setting variable "N" to A, B, C, D or E in sequence. Ex-1008, §8.4.2.1.2. ("the derivation of availableFlagN, with N being A, B, C, D or E"); Ex-1010, §8.4.2.1.2. For each block, including block A, Nakamura evaluates whether the block has "identical motion parameters" with block A. Ex-1008, §8.4.2.1.2 ("PartMode of the current prediction unit is PART_Nx2N and PartIdx is equal to 1 and **the prediction units covering luma location (xP–1, yP) (PartIdx=0)** and luma location (xN, yN) (Cand. N) have identical motion parameters[.]"); Ex-1010, §8.4.2.1.2. Here, the coordinates "(xP-1, yP)" refers to the block to the left of the current block because its x coordinate has been decremented by 1, meaning it is the previous block in the horizontal (x) direction. Since block A is the block to the left of the current block, block A is a PU "covering luma location (xP–1, yP)." *Id.* I have illustrated this below, with a red box showing the block that Nakamura would exclude:



366. Block A has "PartIdx=0" because this is a horizontally-divided block (part mode Nx2N), and block A is to the left of the current block, which has PartIdx=1. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Therefore, when Nakamura iterates to block A, then N is set to A (Ex-1008, §8.4.2.1.2 ("the derivation of availableFlagN, with N being A, B, C, D or E")), and Nakamura excludes block A's potential spatial motion vector prediction candidate from the merge list because, by the law of identity, block A has "identical motion parameters" with itself. *Id.*

367. By setting availableFlagA "equal to 0," block A is not available for, and therefore excluded from, the merging candidate list. Ex-1008, §8.4.2.1.2 ("The merging candidate list, mergeCandList, is constructed of… A, if availableFlagA is equal to 1[.]"); Ex-1010, §8.4.2.1.2.

368. Based on the foregoing explanations, Ground 2 teaches claim 11.

6. Dependent Claim 12

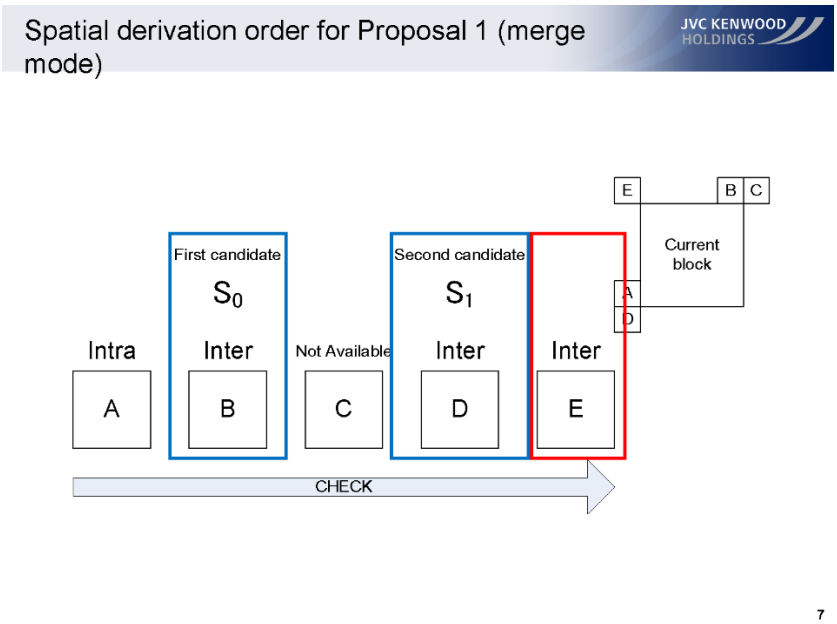
[12pre]	The method according to claim 9 further comprising

369. As I explained above, Ground 2 teaches the method of claim 9. *Supra* §V.B.3.

[12a]	determining a maximum number of spatial motion vector prediction candidates to be included in a merge list; and

370. Ground 2 teaches limitation [12a], as explained below.

371. Ground 2 teaches **determining a maximum number of spatial motion vector prediction candidates to be included in a merge list** in at least two ways. First, Nakamura teaches "[t]he merging candidate list is constructed of two spatial merging candidates[.]" Ex-1007, §2.2.1, Tables 1-2, 4; *supra* §V.B.3[9b]. Therefore Nakamura teaches determining a maximum number of spatial candidates (2) in the merging candidate list because at most two candidates will be included in the list; no additional spatial candidates are added after two. Ex-1009, 000008:



372. As illustrated in the above example, after two spatial candidates S_0 and S_1 (highlighted with blue boxes), corresponding to blocks B and D respectively, are selected for the merging candidate list, block E (in red box) is not included in the merging candidate list even though block E is available for motion vector prediction. Ex-1009, 000008. Therefore, Nakamura teaches that two is the maximum number of spatial candidates in the merging candidate list.

373. Second, Nakamura teaches a maximum number of spatial candidates checked for the merging candidate list. Ex-1008, §8.4.2.1.1 ("If the number of availableFlagM (with M being replaced by A, B, C or D) which is equal to 1 in order of A, B, C, D and E is equal to 4, the availableFlagN is set equal to 0. (Note: If availableFlagA, availableFlagB, availableFlagC and availableFlagD are equal to 1, availableFlagE is set to 0.)"); Ex-1010, §8.4.2.1.2 ("If one of the following

conditions is true, the availableFlagN is set equal to 0,… N is equal to B2 and availableFlagA0 + availableFlagA1 + availableFlagB0 + availableFlagB1 is equal to 4”). Here, Nakamura teaches that if four spatial neighbors are available as spatial candidates for the merging candidate list, then any additional spatial neighbor is set as unavailable, limiting the number of spatial candidates checked for the merging candidate list to four. For example, “[i]f availableFlagA” for block A, “availableFlagB” for block B, “availableFlagC” for block C, and “availableFlagD” for block D “are equal to 1,” indicating that blocks A, B, C, and D are available, then “availableFlagE” for block E “is set to 0.” Ex-1008, §8.4.2.1.1. By limiting the number of spatial candidates that are checked to four, the maximum number of spatial candidates to be included in the merging candidates list is four. Thus, for this additional reason, Nakamura teaches determining a maximum number (e.g., 4) of spatial motion vector prediction candidates to be included in the merge list.

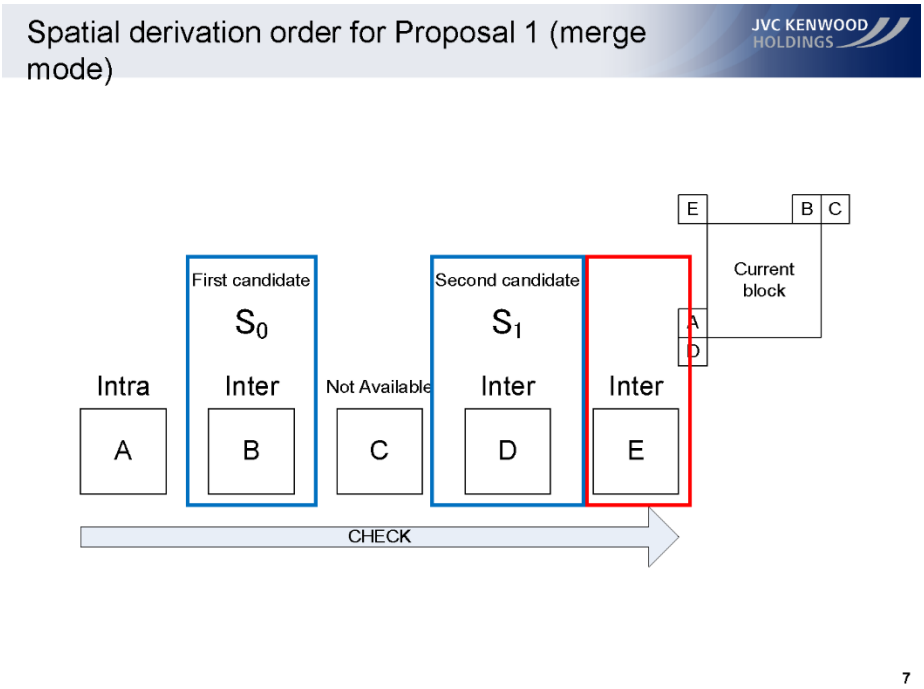
374. Based on the foregoing explanations, Ground 2 teaches limitation [12a].

[12b]	**limiting the number of spatial motion vector prediction candidates in the merge list** [to be] **smaller or equal to the maximum number.**

375. Ground 2 teaches limitation [12b], as explained below.

376. Ground 2 teaches **limiting the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number** at least two ways.

377. First, Nakamura teaches that after two spatial candidates are selected for the merging candidate list, no additional spatial candidates are included in the merging candidate list. Thus, Nakamura limits the number of spatial motion vector prediction candidates in the merging candidate list to be smaller or equal to the maximum number two. *Supra* §V.B.6[12a]; Ex-1007, Tables 3-4, Ex-1009, 000008:



378. As illustrated in the above example, the spatial candidate for block E (in red box) is available for motion vector prediction but not included in the merging candidate list. Ex-1009, 000008. That is because two spatial candidates

(in blue boxes), S_0 and S_1 corresponding to blocks B and D respectively, have already been found. *Id*. Therefore, Nakamura teaches limiting the number of spatial motion vector prediction candidates in the merging candidate list to being smaller or equal to two spatial candidates.

379. Second, Nakamura teaches limiting the number of spatial candidates checked for the merging candidate list to four. Ex-1008, §8.4.2.1.1 ("If the number of availableFlagM (with M being replaced by A, B, C or D) which is equal to 1 in order of A, B, C, D and E is equal to 4, the availableFlagN is set equal to 0. (Note: If availableFlagA, availableFlagB, availableFlagC and availableFlagD are equal to 1, availableFlagE is set to 0.)"); Ex-1010, §8.4.2.1.2 ("If one of the following conditions is true, the availableFlagN is set equal to 0,… N is equal to B2 and availableFlagA0 + availableFlagA1 + availableFlagB0 + availableFlagB1 is equal to 4"); *supra* §V.B.6[12a]. Here, Nakamura teaches that if four spatial neighbors are available as spatial candidates for the merging candidate list, then any additional spatial neighbor is set as unavailable, limiting the number of spatial candidates checked for the merging candidate list to four. For example, "[i]f availableFlagA" for block A, "availableFlagB" for block B, "availableFlagC" for block C, and "availableFlagD" for block D "are equal to 1," indicating that blocks A, B, C, and D are available, then "availableFlagE" for block E "is set to 0." Ex-1008, §8.4.2.1.1. By limiting the number of spatial candidates that are checked

to four, the maximum number of spatial candidates included in the merging candidates list is at most four.

380. Based on the foregoing explanations, Ground 2 teaches limitation [12b].

7. Dependent Claim 13

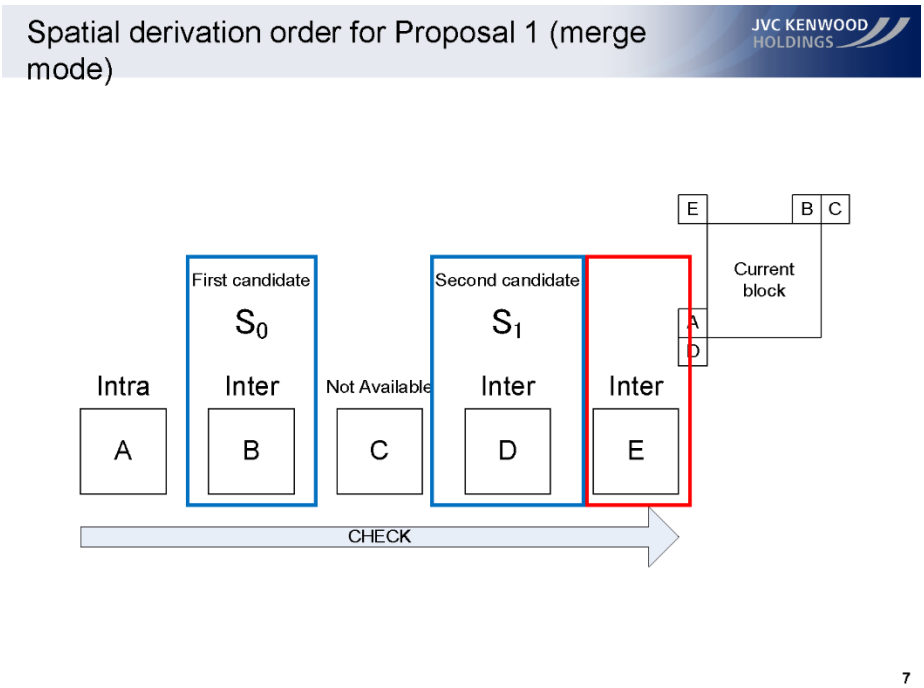
[13pre] The method according to claim 12 comprising:

381. As I explained above, Ground 2 teaches the method of claim 12. *Supra* §V.B.6.

[13a] examining, if the number of spatial motion vector prediction candidates in the merge list [is] **smaller than the maximum number,**

382. Ground 2 teaches limitation [13a].

383. Ground 2 teaches **examining, if the number of spatial motion vector prediction candidates in the merge list smaller than the maximum number**. For example, Nakamura teaches "[t]he merging candidate list is constructed of two spatial merging candidates" out of five spatial neighbors, with two as the maximum number of spatial candidates. Ex-1007, §2.2.1, Tables 1-2, 4; *supra* §V.B.6[12a]. Nakamura further teaches examining if the number of spatial candidates in the merging candidate list is smaller than the maximum number (two) to limit the number to two. *Supra* §V.B.6[12b]; Ex-1007, Table 3, Table 4; Ex-1009, 000008:



384. As illustrated in the above example, the spatial candidate for block D (e.g., S_1) is added when the number of spatial candidates in the merging candidate list is smaller than two, and the spatial candidate for block E is not included in the merging candidate list because the number of spatial candidates in the merging candidates list is no longer smaller than two. The variable "NumMergeCand" represents the "number of elements… within the mergeCandList[.]" Ex-1008, §8.4.2.1.1.

385. Based on the foregoing explanations, Ground 2 teaches limitation [13a].

[13b]	**if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;**

386. Ground 2 teaches limitation [13b], as explained below.

387. Ground 2 teaches **examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction**. For example, Nakamura teaches that "[f]or the derivation of availableFlagN, with N being A, B, C, D or E… the availableFlagN is set equal to 0[,]" or unavailable, "[i]f one of the following conditions is true… [t]he prediction unit… is not available[.]" Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. The prediction unit may not be available, for example, because the prediction unit is intracoded (e.g., "PredMode is MODE_INTRA") or because it does not have an associated motion vector predictor. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Therefore, Nakamura teaches examining whether the PU to which the potential spatial motion vector prediction candidate belongs is available for motion prediction, so as to exclude the candidate if it is not available.

388. Based on the foregoing explanations, Ground 2 teaches limitation [13b].

[13c]	**if so, performing at least one of the following:**
[13d]	**for a potential spatial motion vector prediction candidate on a left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13e]	**the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;**
[13f]	**the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;**
[13g]	**for a potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13h]	**the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;**
[13i]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;**
[13j]	**for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**

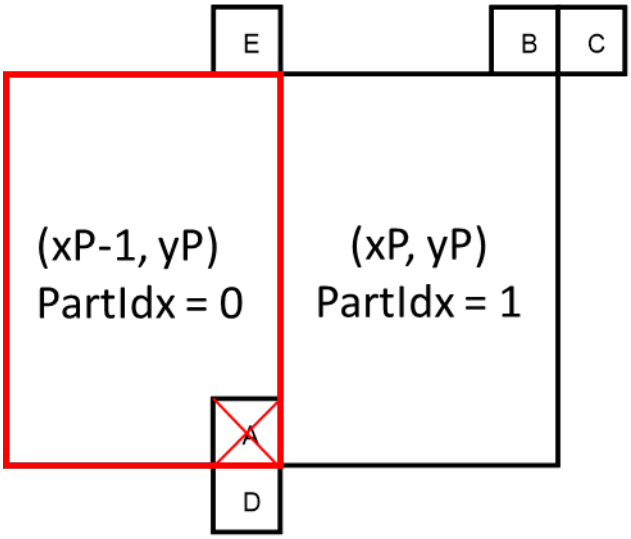
[13k]	**for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and**
[13l]	**for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[13m]	**all the other potential spatial motion vector prediction candidates have been included in the merge list;**
[13n]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**
[13o]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.**

389. Limitation [13c] requires "performing at least one of the following:" and limitations [13d] and [13g] require "if any of the following conditions are fulfilled:" Therefore, I understand that limitations [13c]-[13o] are satisfied if limitations [13d]-[13f] are met. Alternatively, limitations [13c]-[13o] are also satisfied if limitations [13g]-[13h] are satisfied. Ground 2 teaches receiving an encoded block of pixels, which is the current block. Ex-1010, §0.2, §3.11, §3.12, §3.34, §3.38; Ex-1007, §2.2.1, Fig. 2; *supra* §§V.B.3[9a], [9e].

390. Ground 2 teaches limitations [13c]-[13h].

391. For [13d]-[13e], Ground 2 teaches **for a potential spatial motion vector prediction candidate on a left side of the prediction unit for the current block, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled: the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit**. For example, Nakamura excludes the potential spatial motion vector prediction candidate from the merging candidate list by setting the availableFlagN to 0. Nakamura teaches that "[i]f one of the following conditions is true, the availableFlagN is set equal to 0… PartMode of the current prediction unit is PART_Nx2N and PartIdx is equal to 1 and the prediction units covering luma location (xP-1, yP) (PartIdx=0) and luma location (xN, yN) (Cand. N) have identical motion parameters[.]" Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2.

392. Here, for a scenario where the current block of pixels is a Nx2N vertically-divided block (e.g., PartMode is PART_Nx2N) and the PU for the current block is on the right side (e.g., PartIdx=1), Nakamura teaches that the prediction unit on the left side (e.g., PartIdx=0) and any block that has "identical motion parameters" are set as unavailable and therefore excluded from the merging candidate list. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2; *supra* §V.B.5 (explaining HEVC partition modes and ids for claim 11). This scenario is illustrated below:



393. Nakamura iterates through neighboring blocks, setting variable "N" to A, B, C, D or E in sequence. Ex-1008, §8.4.2.1.2 ("the derivation of availableFlagN, with N being A, B, C, D or E"); Ex-1010, §8.4.2.1.2. For each block, including block A, Nakamura evaluates whether the block has "identical motion parameters" with block A. Ex-1008, §8.4.2.1.2 ("PartMode of the current prediction unit is PART_Nx2N and PartIdx is equal to 1 and *the prediction units covering luma location (xP–1, yP) (PartIdx=0)* and luma location (xN, yN) (Cand. N) have identical motion parameters[.]"); Ex-1010, §8.4.2.1.2. Here, the coordinates "(xP-1, yP)" refers to the block to the left of the current block because its x coordinate has been decremented by 1, meaning it is the previous block in the horizontal (x) direction. Since block A is the block to the left of the current block, block A is a PU "covering luma location (xP–1, yP)." *Id.* When Nakamura iterates to block A, N is set to A (Ex-1008, §8.4.2.1.2 ("the derivation of

availableFlag N, with N being A, B, C, D or E”)), and Nakamura excludes block A’s potential spatial motion vector prediction candidate from the merge list because block A has “identical motion parameters” with itself. *Id.*

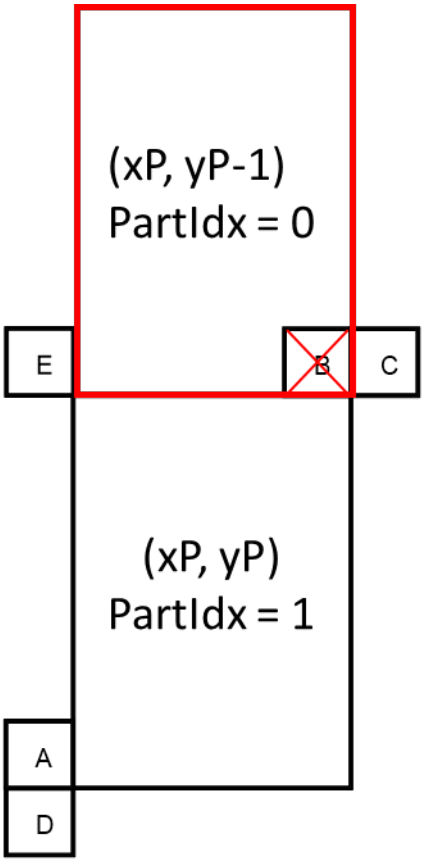
394. In the Nx2N scenario illustrated above, availableFlagA for block A is set “equal to 0,” making block A unavailable for, and therefore excluded from, the merging candidate list. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Block A is excluded because it is “the prediction unit[] covering… (xP-1, yP)” or the block on the left side of the prediction unit in a vertically divided block of pixels. Therefore, for a potential spatial motion vector prediction candidate on a left side of the current PU, Nakamura excludes the potential spatial motion vector prediction candidate from the merging candidate list if the current block is vertically divided into a first and second PU. This exclusions only occurs if the PU on the left (for vertically-divided blocks) is available for motion prediction because otherwise, it would already be marked as unavailable. *Supra* §V.B.7[13b].

395. Therefore, as explained above, Ground 2 teaches limitations [13d]-[13f].

396. For [13g]-[13h], Nakamura teaches **for a potential spatial motion vector prediction candidate above the PU for the current block, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled: the received encoded block of pixels is**

horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit. For example, Nakamura excludes the potential spatial motion vector prediction candidate from the merging candidate list by setting the availableFlagN to 0. Nakamura teaches that "[i]f one of the following conditions is true, the availableFlagN is set equal to 0… PartMode of the current prediction unit is PART_2NxN and PartIdx is equal to 1 and the prediction units covering luma location (xP, yP–1) (PartIdx=0) and luma location (xN, yN) (Cand. N) have identical motion parameters[.]" Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2.

397. Here, for a scenario where a block of pixels is a 2NxN horizontally divided block (e.g., PartMode is PART_2NxN) and the PU for the current block is on the bottom (e.g., PartIdx=1), Nakamura teaches that the prediction unit above the current prediction unit (e.g., PartIdx=0) and any block that has "identical motion parameters" are set as unavailable. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. This scenario is illustrated below:



398. Nakamura iterates through neighboring blocks, setting variable "N" to A, B, C, D or E in sequence. Ex-1008, §8.4.2.1.2 ("the derivation of availableFlag N, with N being A, B, C, D or E"); Ex-1010, §8.4.2.1.2. For each block, including block B, Nakamura evaluates whether the block has "identical motion parameters" with block B. Ex-1008, §8.4.2.1.2 ("PartMode of the current prediction unit is PART_2NxN and PartIdx is equal to 1 and *the prediction units covering luma location (xP, yP–1) (PartIdx=0)* and luma location (xN,yN) (Cand. N) have *identical motion parameters*[.]"); Ex-1010, §8.4.2.1.2. Here, the coordinates "(xP, yP–1)" refers to the block above the current block because it has the same x coordinate but is 1 row up. The y coordinate has 1 subtracted from it, which

results in the previous block in the vertical (y) direction. Since block B is the block above the current block, block B is a PU "covering… (xP, yP-1)." *Id.* When Nakamura iterates to block B, then N is set to B (Ex-1008, §8.4.2.1.2 ("the derivation of availableFlag N, with N being A, B, C, D or E")), and Nakamura excludes block B's potential spatial motion vector prediction candidate from the merge list because block B has "identical motion parameters" with itself. *Id.*

399. In the 2NxN scenario illustrated above, availableFlagB for block B is set "equal to 0," making block B unavailable for, and therefore excluded from, the merging candidate list. Ex-1008, §8.4.2.1.2; Ex-1010, §8.4.2.1.2. Block B is excluded because it is "the prediction unit[] covering… (xP, yP-1)" or the block above the current prediction unit in a vertically divided block of pixels. Therefore, for a potential spatial motion vector prediction candidate above the prediction unit, Nakamura excludes the potential spatial motion vector prediction candidate from the merging candidate list if the received block of pixels is horizontally divided into a first and second PU, and the PU is the second PU. This exclusion only occurs if the PU above (for horizontally-divided blocks) is available for motion prediction because otherwise, it would already be marked as unavailable. *Supra* §V.B.7[13b].

400. Based on the foregoing explanations, Ground 2 teaches limitations [13c]-[13o].

8. Dependent Claim 14

14.	**The method according to claim 9 comprising selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.**

401. Ground 2 teaches claim 14, as explained below.

402. Ground 2 teaches the method according to claim 9 as explained above *supra* §V.B.3.

403. Ground 2 teaches **selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels**. For example, WD4 teaches a merge index (e.g., "merge_idx") that "specifies the merging candidate index of the merging candidate list[.]" Ex-1010, §7.4.7; *supra* §V.B.3[9e]. Based on the merge_idx, one spatial "candidate at position merge_idx[xP][yP] in the merging candidate list mergeCandList" is assigned as the spatial candidate to use for prediction of the current block. Ex-1010, §8.4.2.1.1.

404. As I also explained above with respect to limitation [9b], the merging candidate contains information such as motion vector information and reference picture index, which indicates the motion vector prediction for a prediction unit. *Supra* §V.B.3[9b].

405. Therefore, WD4 teaches selecting one motion vector prediction candidate from the merging candidate list to represent a motion vector prediction

for the current block of pixels. The selected merging candidate is assigned as the spatial candidate to use for prediction of the block of pixels, and therefore, represents the motion vector prediction used for the block of pixels. *Supra* §V.B.3[9g]; Ex-1010, §8.4.2.1.1.

406. Based on the foregoing explanations, Ground 2 teaches claim 14.

9. Independent Claim 16

[16pre] An apparatus comprising a processor and a memory including computer program code, the memory and the computer program code configured to, with the processor, cause the apparatus to:

407. To the extent that the preamble is limiting, Ground 2 teaches limitation [16pre], as explained below.

408. Nakamura teaches an **apparatus comprising a processor and a memory including computer program code**. For example, Nakamura teaches a decoder "for decoding experiments" that comprises a "CPU" and "[m]emory[.]" Ex-1007, Table 6:

Table 6 Simulation environment for decoding experiments

CPU	Intel Core i7 870 2.93GHz
Memory	16GB (DDR3)
OS	Windows 7 Professional 64bit
Compiler	Microsoft Visual C++ 2008 Express edition SP1
Executable	x64

Furthermore, WD4 teaches "video decoders" with various "processing load and memory capability[]" capable of implementing its teachings. Ex-1010, Annex A.

409. Nakamura teaches **the memory and computer program code is configured to, with the processor, cause the apparatus to** perform the recited steps. *Infra* §§V.B.10[18a]-[18g]. For example, Nakamura teaches "[t]his proposed technique is implemented into HM3.0 software[.]" Ex-1007, §2.3.1, *See, e.g.*, Ex-1007, §3.3.3, Tables 7-12 (reporting "[s]imulation results" with decoder runtime). At the very least, it would have been obvious to a POSITA for a video decoder to include a processor and a memory, such as a hard drive, to execute software because conventional computers have included processors and memory for this purpose for decades.

410. Based on the foregoing explanations, Ground 2 teaches the preamble of claim 16.

[16a]	**receive an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;**

411. Ground 2 teaches limitation [16a]. Limitation [16a] is nearly identical to limitation [9a]. *See supra* §V.A.9[16a]. The minor differences do not affect my analysis that, for the same reasons I have discussed for limitation [9a], Ground 2 teaches limitation [16a]. *Supra* §V.B.3[9a].

[16b]	**determine a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

412. Ground 2 teaches limitation [16b]. Limitation [16b] is nearly identical to limitation [9b]. *See supra* §V.A.9[16b]. The minor differences do not affect my analysis that, for the same reasons I have discussed for limitation [9b], Ground 2 teaches limitation [16b]. *Supra* §V.B.3[9b].

[16c]	**select a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

413. Ground 2 teaches limitation [16c]. Limitation [16c] is nearly identical to limitation [9c]. *See supra* §V.A.9[16c]. The minor differences do not affect my analysis that for the same reasons I have discussed for limitation [9c], Ground 2 teaches limitation [16c]. *Supra* §V.B.3[9c].

[16d]	**determine a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

414. Ground 2 teaches limitation [16d]. Limitation [16d] is nearly identical to limitation [9d]. *See supra* §V.A.9[16d]. The minor differences do not affect my analysis that, for the same reasons I have discussed for limitation [9d], Ground 2 teaches limitation [16d]. *Supra* §V.B.3[9d].

[16e]	**compare motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;**

415. Ground 2 teaches limitation [16e]. Limitation [16e] is nearly identical to limitation [9e]. *See supra* §V.A.9[16e]. The minor differences do not affect my analysis that, for the same reasons I have discussed for limitation [9e], Ground 2 teaches limitation [16e]. *Supra* §V.B.3[9e].

[16f]	**exclude the first spatial motion vector prediction candidate from the merge list, if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other; and**

416. Ground 2 teaches limitation [16f]. Limitation [16f] is nearly identical to limitation [9f]. *See supra* §V.A.9[16f]. The minor differences do not affect my analysis that, for the same reasons I have discussed for limitation [9f], Ground 2 teaches limitation [16f]. *Supra* §V.B.3[9f].

[16g]	**select a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

417. Ground 2 teaches limitation [16g]. Limitation [16g] is nearly identical to limitation [9g]. *See supra* §V.A.9[16g]. The minor differences do not affect my analysis that Ground 2 teaches limitation [16g]. *Supra* §V.B.3[9g].

10. Independent Claim 18

[18pre]	**A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprise instructions for use by an encoder, said program code comprises instructions for:**

418. To the extent that the preamble is limiting, Ground 2 teaches limitation [18pre], as explained below.

419. As I explained for Ground 1, the preamble likely contains a typographical error, where "a computer executable program code for use by *a decoder*" is recited as "a computer executable program code for use by *an encoder*." *Supra* §V.A.10[18pre].

420. Based on this understanding, WD4 teaches **a non-transitory computer readable medium having stored thereon a computer executable program code for use by an decoder, said program codes comprising instructions for**. For example, Nakamura teaches "software" by which its

"technique is implemented[.]" Ex-1007, §2.3.1. Nakamura's teachings were applied in software "for encoding and decoding experiments[.]" Ex-1007, §3.2. WD4 teaches an decoder. Ex-1010, §0.5 (addressing the performance of decoders), §3.31 (defining a decoder as "[a]n embodiment of a decoding process"), Annex A (describing the capability of the decoder required to perform video decoding functions). WD4's teachings were tested on software. *See*, *e.g.,* Ex-1010, 000003 (noting minor difference between software and text), §8.3.3.1.8 (noting minor bug in software).

421. To the extent that Patent Owner argues that Nakamura or WD4 does not explicitly state a non-transitory computer readable medium having stored thereon a computer executable program code for use by an decoder, it would have at least been obvious to a POSITA that the video decoder taught in WD4 includes a non-transitory computer-readable medium, such as a hard drive, that stores the computer executable program code comprising instructions for use by a decoder to perform the recited steps. It was well known for a computer, such as one that functions as a decoder in WD4, to include a non-transitory storage medium such as hard drive, where computer codes are stored, in order to perform the recited functionalities.

422. To the extent that the preamble "a computer executable program code for use by *an encoder*" of claim 30 is not a typographical error, Ground 2 also

teaches the preamble. *See* Ex-1010, §0.5 (addressing the performance of encoders), §3.37 (defining an encoder as "[a]n embodiment of an encoding process."). In addition, it would have at least been obvious to a POSITA that the video encoder taught in WD4 includes a non-transitory computer-readable medium, such as a hard drive, that stores the computer executable program code comprising instructions for use by an encoder to perform the recited steps.

423. Based on the foregoing explanations, Ground 2 teaches the preamble of claim 18.

[18a]	**receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;**

424. Ground 2 teaches limitation [18a]. Limitation [18a] is identical to limitation [9a]. For the same reasons I have discussed for limitation [9a], Ground 2 teaches limitation [18a]. *Supra* §V.B.3[9a].

[18b]	**determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;**

425. Ground 2 teaches limitation [18b]. Limitation [18b] is identical to limitation [9b]. For the same reasons I have discussed for limitation [9b], Ground 2 teaches limitation [18b]. *Supra* §V.B.3[9b].

[18c]	**selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;**

426. Ground 2 teaches limitation [18c]. Limitation [18c] is identical to limitation [9c]. For the same reasons I have discussed for limitation [9c], Ground 2 teaches limitation [18c]. *Supra* §V.B.3[9c].

[18d]	**determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;**

427. Ground 2 teaches limitation [18d]. Limitation [18d] is identical to limitation [9d]. For the same reasons I have discussed for limitation [9d], Ground 2 teaches limitation [18d]. *Supra* §V.B.3[9d].

[18e]	**comparing motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;**

428. Ground 2 teaches limitation [18e].

429. Limitation [18e] is substantially similar to limitation [9e]. Their differences are shown below:

[18e]	[9e]
comparing motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;	comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

430. As an initial matter, "*the* spatial motion vector prediction candidate" of limitation [18e] lacks antecedent basis. This limitation is rendered obvious for the same reasons that "*another* spatial motion vector prediction candidate" of limitation [9e] is obvious because both limitations reference a spatial motion vector prediction candidate other than the first spatial motion vector prediction candidate.

431. Furthermore, "*in* the *determined subset*" is rendered obvious for the same reasons that "*of* the *[sub]set*" (incorporating the certificate of correction) is obvious because a comparison between two spatial motion vector prediction candidates in a set teaches a comparison of two spatial motion vector prediction candidates in a subset of the set. Here, the "determined subset" references "a subset" in limitation [18d] which is a subset of "a set" in limitation [18b]. Likewise, "the [sub]set" of limitation [9c] is subset of "a set" in limitation [9b].

432. At the very least, it would be obvious that Ground 2 teaches "comparing … with motion information of the spatial motion vector prediction candidate in the determined subset," because Nakamura teaches comparing the two spatial candidates (S_0 and S_1) to "[r]emove candidates with the same motion information[.]" Ex-1007, Fig. 1, Table 4, Table 2. *See* §V.B.3[9e]. For the reasons I explained for limitation [9e], Ground 2 teaches limitation [18e]. *Supra* §V.B.3[9e].

[18f]	**if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and**

433. Ground 2 teaches limitation [18f]. Limitation [18f] is identical to limitation [9f]. For the same reasons I have discussed for limitation [9f], Ground 2 teaches limitation [18f]. *Supra* §V.B.3[9f].

[18g]	**selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.**

434. Ground 2 teaches limitation [18g]. Limitation [18g] is identical to limitation [9g]. For the same reasons I have discussed for limitation [9g], Ground 2 teaches limitation [18g]. *Supra* §V.B.3[9g].

11. Dependent Claim 26

26.	**The apparatus according to claim 16 wherein the is apparatus is further caused to compare motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.**

435. Ground 2 teaches claim 26. Claim 26 is nearly identical to claim 10. *See supra* §V.A.11. The minor differences do not affect my analysis that, for the same reasons I have discussed for claim 10, Ground 2 teaches claim 26. *Supra* §V.B.4.

12. Dependent Claim 27

27.	**The apparatus according to claim 16 wherein the apparatus is further caused to examine whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, exclude the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.**

436. Ground 2 teaches claim 27. Claim 27 is nearly identical to claim 11. *See supra* §V.A.12. The minor differences do not affect my analysis that, for the same reasons I have discussed for claim 11, Ground 2 teaches claim 27. *Supra* §V.B.5.

13. Dependent Claim 28

[28pre]	**The apparatus according to claim 16 wherein the apparatus is further caused to:**
[28a]	**determine a maximum number of spatial motion vector prediction candidates to be included in a merge list; and**
[28b]	**limit the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number.**

437. Ground 2 teaches claim 28. Claim 28 is nearly identical to claim 12. *See supra* §V.A.13. The minor differences do not affect my analysis that, for the same reasons I have discussed for claim 12, Ground 2 teaches claim 28. *Supra* §V.B.6.

14. Dependent Claim 29

[29pre] The apparatus according to claim 28 wherein the apparatus is further caused to:

[29a] examine if the number of spatial motion vector prediction candidates in the merge list smaller than the maximum number;

[29b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[29c] if so, perform at least one of the following:

[29d] for a potential spatial motion vector prediction candidate on a left side of the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29e] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[29f] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;

[29g] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29h] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;

[29i] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;

[29j]	**for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**
[29k]	**for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and**
[29l]	**for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:**
[29m]	**all the other potential spatial motion vector prediction candidates have been included in the merge list;**
[29n]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;**
[29o]	**the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.**

438. Ground 2 teaches claim 29. Claim 29 is nearly identical to claim 13. *See supra* §V.A.14. The minor differences do not affect my analysis that, for the same reasons I have discussed for claim 13, Ground 2 teaches claim 29. *Supra* §V.B.7.

15. Dependent Claim 30

30.	**The apparatus according to claim 16 wherein the apparatus is further caused to select one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.**

439. Ground 2 teaches claim 30. Claim 30 is nearly identical to claim 14. *See supra* §V.A.15. The minor differences do not affect my analysis that, for the same reasons I have discussed for claim 14, Ground 2 teaches claim 30. *Supra* §V.B.8.

VI. BACKGROUND AND QUALIFICATIONS

440. This section contains a summary of my educational background, career history, publications, and other relevant qualifications. My full curriculum vitae is attached as Appendix 1 to this declaration.

441. I received a bachelor of science degree in Electrical and Computer Engineering from the University of California at Davis in 1985. I received a Master of Science degree in electrical and Computer Engineering from the University of California at Santa Barbara in 1990, and I received my PhD. in Electrical and Computer Engineering, also from the University of California at Santa Barbara in 1993.

442. I have more than 30 years of experience with data compression, decompression, and data storage.

443. I am currently a Full Professor in the Klipsch School of Electrical & Computer Engineering at New Mexico State University. I was an Assistant Professor at New Mexico State from January 2000 until I became an Associate Professor in 2004. I have been a Full Professor since August 2010. My research and coursework at New Mexico State have focused on digital signal and image and video processing. Much of the research I have done over the course of my career is in the area of image and video compression.

444. My first exposure to the field of signal compression came in the fall of 1989, when I took a course entitled Vector Quantization and Signal Compression at UCSB from Prof. Allen Gersho—an internationally renowned researcher in the area of speech compression. As my Ph.D. research progressed, I began to focus on transform-based compression as my main application area. My first paper dealing with image compression was published in 1991. I have since written 24 other journal and conference papers directly related to compression, and I am the named inventor on two issued United States patents related to compression.

445. Since joining the faculty of New Mexico State University in 2000, I have taught numerous classes at both the graduate and undergraduate levels. At the graduate level, I have taught the following: Signal Compression (EE573), Image Processing (EE596), Digital Signal Processing (EE545), Pattern Recognition (EE565), Advanced Linear Systems (EE555), Telemetering Systems

(EE585), Information Theory (EE586), Adaptive Signal Processing (EE594), Multirate Signal Processing and Wavelets (EE595), and Neural Signal Processing (EE590). At the undergraduate level, I have taught the following courses: Linear Algebra & Probability (EE200), Signals and Systems I (EE312), Image Processing (EE446), Introduction to Digital Signal Processing (EE395), and Digital Communications (EE497).

446. From 1993 through 1999, I was a Researcher and Team Leader, at the Naval Air Warfare Center, China Lake. At China Lake, my research efforts focused on high speed image and video compression technologies including embedded wavelet video compression. I also developed a real-time video streaming system that efficiently operated over TCP/IP networks while retaining the highest possible fidelity.

447. From 1990 through 1993, I worked as a Research Assistant in the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. In this position, I worked on subband coding (compression) and multirate filter bank theory. I also implemented real-time filter banks on a digital signal processer. In the summer of 1992, I worked at AT&T Bell labs where I developed and simulated new methods of extremely low bit rate video coding for video telephone applications.

448. From 1985 through 1989, I worked as a Design Engineer at the Naval Weapons Center, China Lake. In this role, I built and tested the guidance electronics for various laser guided munitions. This project included mixed analog and digital circuit design as well as the programming of an embedded digital signal processor. I also developed software for an advanced video processor and studied ground target tracking.

449. A full listing of my publications is found in my curriculum vitae. *See* Appendix 1.

450. I have published 17 peer-reviewed journal articles and 94 conference papers, including 9 journal articles and 30 conference papers directly related to data compression; the following are representative:

- C.D. Creusere, "A new method of robust image compression based on the embedded zerotree wavelet algorithm," IEEE Trans. on Image Processing, Vol 6, No. 10, Oct. 1997, pp. 1436-1442.
- C.D. Creusere, "Fast embedded compression for video," IEEE Trans. on Image Processing, Vol. 8, No. 12, pp. 1811-16, December 1999.
- C.D. Creusere, "Motion compensated video compression with reduced complexity encoding for remote transmission," Signal Processing: Image Communications, Vol. 16, pp. 627-42, April 2000

451. I am a named co-inventor on two issued patents, both relating specifically to image and video compression. I am the listed inventor on U.S. Patent No. 6,148,111 entitled "Parallel digital image compression system which exploits zerotree redundancies in wavelet coefficients" and U.S. Patent No. 6,466,698 entitled "Efficient embedded image and video compression using lifted wavelets."

452. In addition to the experience and publications listed above, I have also received the following awards and distinctions that are relevant to the subject matter of this declaration. I am currently deputy Editor-in-Chief for IEEE Transactions on Image Processing as well as an Area Editor for IEEE Open Journal on Signal Processing. I have previously served as an Associate Editor for IEEE Transactions on Image Processing from 2010 through 2014. I have also served in this capacity from 2002 through 2005. From 2008-2013, I served as an Associate Editor for IEEE Transactions on Multimedia. I also served as a Senior Area Editor for IEEE Transactions on Image Processing from 2016-2022.

453. In 2004, I served as the co-general chair for the IEEE Digital Signal Processing Workshop in Taos, New Mexico. In 2012 and 2014, I served as the co-technical chair for the Southwest Symposium on Image Analysis and Interpretation held in Santa Fe, New Mexico and San Diego, CA, respectively. In addition, I served as the technical chair for the 2015 and 2021 International Telemetering

Conference held in Las Vegas, NV in October. I am also a member of the technical program committees for the IEEE Data Compression Conference, IEEE International Conference on Image Processing and the IEEE Acoustics, Speech, and Signal Processing Conference.

A. Compensation

454. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate for this type of consulting activity. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

B. Materials and Other Information Considered

455. I have considered information from various sources in forming my opinions. I have reviewed and considered each of the exhibits listed in the attached Appendix 2 (Materials Considered in the Preparation of This Declaration) in forming my opinions.

VII. UNDERSTANDING OF THE LAW

456. I am not an attorney. In forming my opinions in this Declaration, I applied the relevant legal principles provided to me by counsel, which are summarized in Appendix 4.

VIII. RESERVATION OF RIGHTS

457. My opinions are based upon the information that I have considered to date. I am unaware of any evidence of secondary considerations with respect to

the '833 patent that would render any of the challenged claims non-obvious. I reserve the right, however, to supplement my opinions in the future to respond to any arguments raised by the owner of the '833 patent and to take into account new information that becomes available to me.

458. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed on October 16, 2024.

By:

Charles D. Creusere

APPENDIX 1: CURRICULUM VITAE OF CHARLES D. CREUSERE

VITA

CHARLES D. CREUSERE

Klipsch School of Electrical & Computer Engineering
Mailing Address:
MSC 3-0
New Mexico State University
P.O. Box 30001
Las Cruces, NM 88003-8001
Phone: 575-646-3919
Fax: 575-646-1435

email: ccreuser@nmsu.edu

DISSERTATION TITLE

"Perfect Reconstruction Modulated Polyphase Filter Banks Using Reverse-Time Subfilters."

ACADEMIC TRAINING

1980-1985: University of California at Davis, B.S. in Electrical and Computer Engineering.

1989-1990: University of California at Santa Barbara, M.S. in Electrical and Computer Engineering.

1990-1993: University of California at Santa Barbara, Ph.D. in Electrical and Computer Engineering.

PROFESSIONAL EXPERIENCE

2010-Present Holder of the Frank Carden Endowed Chair in Telemetering & Telecommunications and Full Professor. Current research interests include compressive sensing/sparse reconstruction for LIDAR and streaming sensor data as well as EEG brain analysis for audiovisual perceptual quality assessment and modeling.

October 2008 Selected for the International Foundation for Telemetering Endowed Professorship.

Jan. 2000-2008 Assistant/Associate professor in the Klipsch School of Electrical & Computer Engineering. My teaching areas include digital signal processing, image processing, pattern classification, and source coding (signal compression). I have done past research in areas of image, video, and audio compression as well as feature vector extraction for pattern classification. Currently, my research interests include distributed compression, polarimetric image processing for scene analysis, and nonstationary signal denoising.

1993-1999: Researcher & Team Leader, Naval Air Warfare Center, China Lake. My research efforts have focused on high speed image and video compression technologies which offer unique capabilities such as robustness to transmission errors and regional localization. My team (2 other people) and I have implemented a real-time (3 to 15 frames/second with 240x512 frames)

320C80-based system which uses a wavelet transform along with embedded coding techniques to compress a video input and stream it through the Internet via TCP/IP protocols. Our recent research focus has been to add more intelligence to the encoder so that the space-frequency information in the image that is most useful for image analysis is received with the highest fidelity. While most of my recent research has been in the area of embedded compression, I am still very much interested in other applications of time/space-frequency decompositions and of multirate digital signal processing concepts in general.

1999, Spring Quarter: Instructor at the University of California at Santa Barbara. Taught graduate class in Multirate Digital Signal Processing, ECE 258B.

1990-1993: Research Assistant, Department of Electrical and Computer Engineering, University of California, Santa Barbara. Worked under Prof. S.K. Mitra on subband coding and multirate filter bank theory. Also implemented real-time filter banks on a Motorola 56001 digital signal processor.

1992: Summer Employee, AT&T Bell labs, Murray Hill, NJ. Developed and simulated new methods of extremely low bit rate video coding for video telephone applications.

1985-1989: Design Engineer, Naval Weapons Center, China Lake. Designed, built, and tested the guidance electronics for the Laser Guided Training Round. This project included mixed analog and digital circuit design as well as the programming of an embedded DSP. Also developed software for an advanced video processor and studied ground target tracking.

FUNDED RESEARCH

• (2000) Office of Naval Research, *Compression of Digital Elevation Maps Using Non-linear Wavelets*, 2000-2003, $94K

• (2001) Sandia National Labs, *Intelligent Compression for Remote Sensing*, 2001-2003, $70K.

• (2002) National Science Foundation (Early Career Grant), *Efficient Audio Compression with Perceptually Embedded Scalability*, 2002-2007, $350K.

• (2004) National Geospatial-Intelligence Agency, *Passive Polarimetric Imagery Classification Study*, 2004-2006, $160K (joint with Dr. David Voelz).

• (2005) Los Alamos National Laboratories, *Signal Detection via Adapted Filter Banks and Geometric Dimensionality Reduction*, 2005-2006, $15K (unburdened).

• (2006) Los Alamos National Laboratories, *Signal Detection via Adapted Filter Banks and Geometric Dimensionality Reduction*, 2006-2007, $50K (unburdened).

• (2006) National Geospatial-Intelligence Agency, *Exploiting Polarization in Imaging Systems*, 2006-2009, $304K (joint with Dr. David Voelz).

• (2006) Army Research Office, *Distributed Source Coding Using Bitstream-based Detection and Classification*, 2006-2009, $326K.

• (2006) DARPA (Subcontract from LANL), *ADAM Project*, 2006-2007, $104K (joint with Dr. Joe Lakey and Dr. Jaime Ramirez)

• (2009) NMSU IRG, *Perceptual audio quality evaluation by direct measurement of human brain responses*, 2009-2010, $39K (joint with Dr. Jim Kroger, Psychology)

• (2011) National Science Foundation, *CIF:Medium:Assessment and modeling of temporal variation in perceived audio and video quality using direct brainwave measurement*, 2011-2015, $917K (lead PI with Dr. Jim Kroger and Dr. Joerg Kliewer as co-PIs)

• (2011) NASA EPSCOR, *Proximity Operations for Near Earth Asteroid Exploration,* 2011-2014, $750K (co-PI, with Dr. Eric Butcher (lead), others)

• (2012) National Geospatial Intelligence Agency (NGA), *Pulse Complexity Based LIDAR Scene Modeling for Sparse Reconstruction and Super-Resolution*, 2012-2013 (plus 3 1 year options), $150K ($75K/option year), co-PI Dr. David Voelz.

• (2018) Airforce Research Lab (AFRL), Software Radio Design in LabView FPGA, 2018-2019, $140K.

PATENTS

• Patent titled "Parallel digital image compression system which exploits zerotree redundancies in wavelet coefficients," Patent Number 6,148,111.

• Patent titled "Efficient embedded image and video compression using lifted wavelets," Number: 6,466,698, granted October 15, 2002.

OTHER DISTINCTIONS

• Awarded the International Foundation for Telemetering Professorship, October 2008.

• Received an educational fellowship from the Department of Defense, 1989-1992.

• Certificate of Merit for the outstanding technical paper awarded at the AIAA Missile Sciences Conference for the paper “Automatic target recognition directed image compression,” Nov. 1998.

• Patent (classified) “Notice of Allowability” titled, "Microcontroller-Based Laser Pulse Decoder," granted October 7, 1991.

• Associate editor for IEEE Trans. on Image Processing, 2002-2005, 2010-2014

• Associate editor for IEEE Trans. on Multimedia , 2008-2013.

• Guest Editor, “Issue on Advances in Hyperspectral Data Processing and Analysis”, IEEE Journal of Selected Topics in Signal Processing, Vol. 5, Numbers: 5 & 6, August-September 2015,

• Co-general chair, IEEE Digital Signal Processing Workshop, August 2004, Taos, NM.

• Co-technical chair for the 2012 and 2014 Southwest Symposium on Image Analysis and Interpretation.

• Student Paper Contest Chair, 40th Asilomar Conf. on Signals, Systems, and Computers, October 2006.

- Organized special session entitled "Applications of Multirate DSP" at the 40th Asilomar Conf. on Signals, Systems, and Computers, October 2006.
- Member of technical program committees for the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), the IEEE International Conference on Image Processing (ICIP), and the IEEE Data Compression Conference (DCC).
- Senior Area Editor, IEEE Transactions on Image Processing, March 2016 to May 2022.
- Area Editor for IEEE Access, June 2022-Present.
- Deputy Editor-in-Chief, IEEE Transactions on Image processing, May 2023-present.

CONSULTING ACTIVITIES

- Video compression systems (technology consultant), Abba Tech, Albuquerque, NM, 2000.

- Expert witness in laser rangefinding technology, Asia Optical Inc. (through NY law firm of Osterlenk, Faber, Gerb & Soffen), Case: LTI versus Nikon/AOI, July 2001-2003. Case went to trial/ testified in court.

- Technical expert for defense; Case: Real-Time v. AT&T (byte.mobile), 2011-2012, case settled June 2012.

- Technical expert for defense; Case: Princeton Digital v. Dell, 2014-2015, case dismissed June 2015.

- Technical expert for defense; Noninfrigement & IPR (6,597,812), Real-time v. SAP, 1/2016-6/2016.

- Technical expert for defense; IPRs (7,378,992 & 8,643,513), Real-time v. Riverbed, et al., 2/2016-2017

- Technical expert for defense; IPRs (7,415,530, 9,116,908, 7,161,506, & 9,054,728), Real-time v. Dell, et al., 2/2016-2017

- Technical expert for defense; Noninfringement, Real-time v. HP Enterprises, 4/2016-2018.

- Technical expert for defense; IPR (7,358,867, 7,161,506, & 9,054,728), Real-time v. Teradata 11/2016-2017

- Technical expert for defense; IPRs (7,415,530, 8,643,513 & 7,378,992), Real-time v. Veritas 12/2016-2017.

- Technical expert for defense; IPRs (7,075,917 & 6,304,612), UNILOC v. Apple 9/2018-2019.

- Technical expert for defense; IPR (7,558,730), Advanced Voice Recognition Systems v. Apple 7/2019-2020.

- Technical expert for defense; District Court; Noninfringement, Realtime Adaptive Systems v. YouTube/Google, 2018-2020.

- Technical expert for defense; ITC case; Nokia v. Lenovo, Oct. 2020-2021.

- Technical expert for defense; IPR (10,176,848), Maxell v. Apple, 2019-2021.

- Technical expert for plaintiff (rebutting invalidity); East Texas District Court; USAA v. PNC Bank, 2021-present. Deposed in Case 1 and Case 2; Attended trial in May 2022 for Case 1 and September 2022 for Case 2 but was not called to testify (USAA won on all counts, patents upheld as valid); I have also opined as an expert in 3 related IPRs (deposed in one of those cases, so far).

- Technical expert for defense; District Court; Gesture Tech Partners v. Apple/Lenovo&Motorola, 2021.

JOURNAL PUBLICATIONS

1. **C.D. Creusere and S.K. Mitra**, "A simple method for designing high-quality prototype filters for M-band pseudo-QMF banks," *IEEE Trans. on Signal Processing*, Vol. 43, No. 4, April 1995, pp. 1005-1007.

2. **C.D. Creusere and S.K. Mitra**, "Efficient audio coding using perfect reconstruction noncausal IIR filter banks," *IEEE Trans. on Speech and Audio Processing,* Vol. 4, No. 2, March 1996, pp. 115-123.

3. **C.D. Creusere and S.K. Mitra**, "Image coding using wavelets based on perfect reconstruction IIR filter banks," *IEEE Trans. on Circuits and Systems for Video Technology,* Vol. 6, No. 5, Oct. 1996, pp. 447-458.

4. **C.D. Creusere**, "A new method of robust image compression based on the embedded zerotree wavelet algorithm," *IEEE Trans. on Image Processing*, Vol 6, No. 10, Oct. 1997, pp. 1436-1442.

5. **C.D. Creusere and A. Van Nevel**, "ATR-directed image and video compression," *Journal of Aircraft,* Vol. 36, No. 4, pp. 626-31, July-August 1999.

6. **C.D. Creusere**, "Fast embedded compression for video," *IEEE Trans. on Image Processing*, Vol. 8, No. 12, pp. 1811-16, December 1999.

7. **C.D. Creusere**, "Motion compensated video compression with reduced complexity encoding for remote transmission," *Signal Processing: Image Communications,* Vol. 16, pp. 627-42, April 2000.

8. **C.D. Creusere**, "Understanding perceptual distortion in MPEG scalable audio coding," *IEEE Trans. on Speech and Audio Processing,* Vol. 13, No. 3, pp. 422-431, May 2005.

9. **L. E. Boucheron and C.D. Creusere**, "Lossless wavelet-based compression of digital elevation maps for fast and efficient search and retrieval," *IEEE Trans. on Geoscience and Remote Sensing,* Vol. 43, No. 5, pp. 1210-1214, May 2005.

10. **V. Thilak, D. Voelz, and C.D. Creusere**, "Polarization-based index of refraction and reflection angle estimation for remote sensing applications," *Applied Optics*, Vol. 46, Bo. 30, pp. 7427-7536, Oct. 2007.

11. **C.D. Creusere, K. Kallakuri, and R. Vanam**, "An Objective Metric of Human Subjective Audio Quality Optimized for a Wide Range of Audio Fidelities," *Audio, Speech, and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on]* , vol.16, no.1, pp.129-136, Jan. 2008

12. **S. Kandadai and C.D. Creusere**, "Scalable Audio Compression at Low Bitrates," *Audio, Speech, and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on]* , vol.16, no.5, pp.969-979, July 2008

13. **S. Kandadai and C.D. Creusere**, "Reverse engineering and repartitioning vector quantizers using training set synthesis," *Signal Processing,* August 2008.

14. **V. Thilak, C.D. Creusere, and D. Voel**z, "Passive Polarimetric Imagery-Based Material Classification Robust to Illumination Source Position and Viewpoint," *Image Processing, IEEE Transactions on* , vol.20, no.1, pp.288-292, Jan. 2011.

15. **C.D. Creusere and J. Hardin**, "Assessing the Quality of Audio Containing Temporally Varying Distortions," *Audio, Speech, and Language Processing, IEEE Transactions on* , vol.19, no.4, pp.711-720, May 2011.

16. **Castorena, J.; Creusere, C.D.**, "The Restricted Isometry Property for Banded Random Matrices," *Signal Processing, IEEE Transactions on* , vol.62, no.19, pp.5073-5084, Oct.1, 2014 doi: 10.1109/TSP.2014.2345350.

17. **Castorena, J.; Creusere, C.D.**, "Sampling of Time-Resolved Full-Waveform LIDAR Signals at Sub-Nyquist Rates," *Geoscience and Remote Sensing, IEEE Transactions on* , vol.53, no.7, pp.3791-3802, July 2015. doi: 10.1109/TGRS.2014.2383839.

REFEREED CONFERENCE PUBLICATIONS

1. **H. Babic, S.K. Mitra, C.D. Creusere, and A. Das**, "Perfect reconstruction recursive QMF banks for image subband coding," *Proc. Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1991, pp. 746-750.

2. **S.K. Mitra, C.D. Creusere, and H. Babic**, "A novel implementation of perfect reconstruction QMF banks using IIR filters," *Proc. IEEE Int. Symposium on Circuits and Systems*, San Diego, CA, May 1992, pp. 2312-2315.

3. **S.K. Mitra, C.D. Creusere, and H. Babic**, "Design of transmultiplexers using IIR filter banks," *Signal Processing VI: Theories and Applications*, Elsevier Science Publishers, 1992, pp. 223-226.

4. **C.D. Creusere and S.K. Mitra**, "Efficient image scrambling using polyphase filter banks," *Proc. International Conference on Image Processing*, Austin, TX, Nov. 1994, pp. 81-85.

5. **C.D. Creusere and G. Hewer**, "Wavelet-based nearest neighbor pattern classification using scale sequential matching," *Proc. Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, Nov. 1994, pp. 1123-1127.

6. **C.D. Creusere**, "Embedded zerotree image coding using low complexity IIR filter banks," *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, Detroit, MI, May 1995, pp. 2213-16.

7 **C.D. Creusere and Gary Hewer**, "Digital video compression for weapons control and bomb damage indication," *AGARD Conference Proceedings 576*, Chapter 16, Sept. 1995.

8. **C.D. Creusere**, "Image coding using parallel implementations of the embedded zerotree wavelet algorithm," *Proc. of the Digital Video Compression Conference (Algorithms and Technologies 1996)*, San Jose, CA, Jan. 28-Feb. 2, 1996, pp. 82-93.

9. **C.D. Creusere**, "A family of image compression algorithms which are robust to transmission errors," *Proceedings of the SPIE*, Vol. 2825, Denver, CO, August, 1996, pp. 890-900.

10. **C.D. Creusere**, "Perfect reconstruction time-varying IIR filter banks," *Conf. Rec. Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1996, pp. 1319-23.

11. **C.D. Creusere**, "Out-of-loop motion compensation for reduced complexity video encoding," *Proc. of the Data Compression Conf.* (pp. 428) *& Data Compression Industry Workshop* (pp.28-37), March 1997, Snowbird, UT.

12. **C.D. Creusere**, "Periodic pan compensation for reduced complexity video compression," *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing,* Vol IV, pp. 2889-92, April 1997, Munich, Germany.

13. **C.D. Creusere**, "A new approach to global motion compensation which reduces video encoding complexity," *Proc. Int. Conf. on Image Processing*, Vol. III, pp. 634-7, October 1997, Santa Barbara, CA.

14. **C.D. Creusere**, "Spatially partitioned lossless image compression in an embedded framework," *Conf. Rec. 31st Asilomar Conf. on Signals, Systems, and Computers*, Nov. 1997, Pacific Grove, CA.

15. **C.D. Creusere**, "Adaptive embedding for reduced complexity image and video compression," *Proc.*

of the SPIE, Vol 3309 (Visual Communications and Image Processing), pp. 48-57, Jan. 1998, San Jose, CA.

16. **C.D. Creusere**, "Successive coefficient refinement for embedded lossless image compression," *Proc. of the Data Compression Conf.*, pp. 539, March 1998, Snowbird, UT.

17. **C.D. Creusere**, "Subband coding of speech and audio," *Proc. of the European Signal Processing Conf.* (invited paper), Sept. 1998, Isle of Rhodes, Greece.

18 **C.D. Creusere**, "Fast embedded video compression using cache-based processing," *Proc. of the European Signal Processing Conf.*, Sept. 1998, Isle of Rhodes, Greece.

19. **C.D. Creusere**, "Successive coefficient refinement for embedded lossless image compression," *Proc. Int. Conf. on Image Processing*, Oct. 1998, Chicago, IL.

20. **C.D. Creusere** and A. Van Nevel, "Autonomous target recognition directed image compression," *Proc. of the AIAA*, Nov. 1998.

21. **C.D. Creusere**," Improved successive refinement for wavelet-based embedded image compression," *Proc. of the SPIE*, Denver, CO, July 1999.

22. A. Van Nevel and **C.D. Creusere**, "Intelligent Bandwidth Compression," *Proc. of the SPIE*, Denver, CO, July 1999.

23. **C.D. Creusere,** "Optimal refinement/significance map tradeoffs in SPIHT-based image compression," *Conf. Rec., 34th Asilomar Conf. on Signals, Systems, & Computers, pp. 1026-30, Oct. 2000.*

24. **C.D. Creusere,** "Compression of digital elevation maps using nonlinear wavelets," *Proc. Int. Conf. on Image Processing*, pp. 824-7, October 2001.

25. **C.D. Creusere and G. Dahman**, "Object detection and localization in compressed video," *Conf. Rec. 35th Asilomar Conf. on Signals, Systems, and Computers*, Nov. 2001, Pacific Grove, CA.

26. **C.D. Creusere**, "An analysis of perceptual artifacts in MPEG scalable audio coding," *Proceedings of the Data Compression Conference*, pp. 152-161, April 2002, Snowbird, UT.

27. **C.D. Creusere and N. Tolk**, "Combining wavelets and GLICBAWLS to achieve resolution-progressive lossless compression," *Proc. of the International Conference on Image Processing*, pp. III-229-32, October 2002.

28. **L. Boucheron and C.D. Creusere**, "Compression of digital elevation maps for fast and efficient search and retrieval," *Proc. of the International Conference on Image Processing*, pp. 629-32, September 2003.

29. **S. Kandadai and C.D. Creusere**, "An experimental study of object detection in the wavelet domain," *Conf. Rec. 37th Asilomar Conf. on Signals, Systems, and Computers*, pp. 1620-4, Nov. 2003, Pacific Grove, CA.

30. **C.D. Creusere**, "Quantifying perceptual distortion in scalably compressed MPEG audio," *Conf. Rec. 37th Asilomar Conf. on Signals, Systems, and Computers*, pp. 265-9, Nov. 2003, Pacific Grove, CA.

31. **C.D. Creusere and L. Zhou**, "Spatial object detection and classification in JPEG bitstreams ," *Proceedings 11th Digital Signal Processing Workshop*, pp. 115-9, August 2004, Taos Ski Valley, NM,

32. **L. Zhou and C.D. Creusere**, "Spatial object detection in JPEG bitstreams," *Proceedings European Conference on Signal Processing*, pp. 949-52, September 2004, Vienna, Austria.

33. **V. Thilak and C.D. Creusere**, "Tracking of extended size targets in H.264 compressed video using the probabilistic data association filter," *Proceedings European Conference on Signal Processing*, pp. 281-4, September 2004, Vienna, Austria.

34. **S. Kandadai and C.D. Creusere**, Reverse engineering vector quantizers by training set synthesis," *Proceedings European Conference on Signal Processing*, pp. 789-92, September 2004, Vienna, Austria.

35. **V. Thilak, J. Saini, D. G. Voelz, C. D. Creusere** , "Pattern recognition for passive polarimetric data using nonparametric classifiers," *Proc. of the SPIE Vol. 5888*, p. 337-344, Polarization Science and Remote Sensing II; Joseph A. Shaw, J. Scott Tyo; Eds.

36. **R. Vanam and C.D. Creusere**, "Evaluating low bitrate scalable audio quality using advanced version of PEAQ and energy equalization approach ," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vol. III, pp. 189-92, March 2005.

37. **S. Kandadai and C.D. Creusere**, "Reverse engineering vector quantizers for repartitioned signal spaces," *Proc. 39th Asilomar Conference on Signals, Systems, and Computers*, pp. 1208-12, Pacific Grove, CA, Nov. 2005.

38. **R. Vanam and C.D. Creusere,** "Scalable perceptual metric for evaluating audio quality," *Proc. 39th Asilomar Conference on Signals, Systems, and Computers*, pp. 319-23, Pacific Grove, CA, Nov. 2005.

39. **S. Kandadai and C.D. Creusere**, "Perceptually-weighted audio coding that scales to extremely low bitrates," *Proc. IEEE Data Compression Conference*, pp. 382-391, Snowbird, UT, March 2006.

40. **V. Thilak, D. Voelz, C. Creusere, S. Damarla,** "Estimating the refractive index and reflected zenith angle of a target using multiple polarization measurements," Proc. SPIE Vol. 6240, 624004, Polarization: Measurement, Analysis, and Remote Sensing VII; Dennis H. Goldstein, David B. Chenault; Eds., May 2006.

41. **V.M. Prasad and C.D. Creusere**, "Analyzing reversible lapped transformations using Reng probing," *Proc. 40th Asilomar Conference on Signals, Systems, and Computers*, pp. 873-877, Oct. 2006.

42. **N. Balachandran and C.D. Creusere**, "Chirp classification using hidden Markov models," *Proc. 40th Asilomar Conference on Signals, Systems, and Computers*, pp. 545-549, Oct. 2006.

43. **V. Thilak and C.D. Creusere**, "Estimating the complex index of refraction and view angle of an object using multiple polarization measurements," *Proc. 40th Asilomar Conference on Signals, Systems, and Computers*, pp. 1067-1071, Oct. 2006.

44. **V. Thilak, C.D. Creusere, and D.G. Voelz**, "Material classification using passive polarmetric imagery," Proc. IEEE Int. Conf. on Image Proc., Vol. 4, pp. IV-121-124, Sept., 2007.

45. **A. Pamba, V. Thilak, D. G. Voelz, C. D. Creusere** "Estimation of incidence and reflection angles from passive polarimetric imagery: extension to out-of-plane scattering," Proc. SPIE Vol 6682, 66820O, Polarization Science and Remote Sensing III, Joseph A. Shaw; J. Scott Tyo, Editors, September 2007.

46. **V. Thilak, D. G. Voelz, C. D. Creusere**, "Image segmentation from multi-look passive polarimetric imagery, Proc. SPIE 6682, 668206, Polarization Science and Remote Sensing III, Joseph A. Shaw; J. Scott Tyo, Editors, October 2007.

47. **S. Kandadai and C.D. Creusere**, "Optimal Bit Layering for Scalable Audio Compression Using Objective Audio Quality Metrics," *Signals, Systems and Computers, 2007. ACSSC 2007. Conference Record of the Forty-First Asilomar Conference on* , pp.560-564, 4-7 Nov. 2007.

48. **C.D. Creusere and I. Mecimore**, "Bitstream-based overlap analysis for multi-view distributed video coding," *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 93-96, March 2008.

49. **V. Thilak, C.D. Creusere, and D.G. Voelz**, "Passive Polarimetric Imagery Based Material Classification For Remote Sensing Applications," *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 153-156 March 2008.

50. **V. Thilak, Q. Wang, D. G. Voelz, C. D. Creusere**, "Estimation of target geometry from Mueller matrix imagery," *Proc. SPIE 6972*, March 2008.

51. **C.D. Creusere and I. Mecimore**, "Bitstream-based correlation detector for multi-view distributed video coding applications," *Proc. Int. Conf. on Accoustics, Speech, and Signal Processing*, pp. 1001-1004, April 2008, Las Vegas, NV.

52. **S. Kandadai, J. Hardin, and C.D. Creusere**, "Audio quality assessment using the mean structural similarity measure," *Proc. Int. Conf. on Accoustics, Speech, and Signal Processing*, pp. 221-224, April 2008, Las Vegas, NV.

53. **S. Matta and C.D. Creusere**, "Efficient correlation extraction for distributed audio coding," *Signals, Systems and Computers, 2008 42nd Asilomar Conference on* , vol., no., pp.1272-1276, 26-29 Oct. 2008 .

54. **J.C. Hardin and C.D. Creusere**, "Objective analysis of temporally varying audio quality metrics," *Signals, Systems and Computers, 2008 42nd Asilomar Conference on* , vol., no., pp.1245-1249, 26-29 Oct. 2008 .

55. **S. Matta and C.D. Creusere**, "Distributed audio coding with efficient source correlation extraction," *Proceedings 13th Digital Signal Processing Workshop*, pp. 16-20, January 2009, Marco Island, FL.

56. **I. Mecimore and C.D. Creusere**, "Unsupervised bitstream based segmentation of images," *Proceedings 13th Digital Signal Processing Workshop*, pp. 642-647, January 2009, Marco Island, FL.

57. **Q. Wang, C.D. Creusere, V. Thilak, and D.G. Voeltz**, "Active polarimetric imaging for estimation of scene geometry," *Proceedings 13th Digital Signal Processing Workshop*, pp. 659-663, January 2009, Marco Island, FL.

58. **J.C. Hardin and C.D. Creusere**, "A temporally varying objective audio quality metric," *Proceedings 13th Digital Signal Processing Workshop*, pp. 21-25, January 2009, Marco Island, FL.

59. **I. Mecimore, W. Fahrenkrog, and C.D. Creusere**, "On bitstream based edge detection techniques," ITEA Conference, January 2009, El Paso, TX (best graduate student paper award).

60. **Mecimore, Ivan; Creusere, Charles D.**; , "Low complexity multi-view distributed video coding based on JPEG," *Image Analysis & Interpretation (SSIAI), 2010 IEEE Southwest Symposium on* , vol., no., pp.165-168, 23-25 May 2010.

61. **Creusere, C.D.; Mehta, K.; Voelz, D.G.**; , "Model-based estimation of surface geometry using passive polarimetric imaging," *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International* , vol., no., pp.4557-4560, 25-30 July 2010.

62. **Castorena, J.; Creusere, C.D.; Voelz, D.**; , "Modeling lidar scene sparsity using compressive sensing," *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International* , vol., no., pp.2186-2189, 25-30 July 2010.

63. **Castorena, J.; Creusere, C.D.; Voelz, D.**; , "Using finite moment rate of innovation for LIDAR waveform complexity estimation," *Signals, Systems and Computers (ASILOMAR), 2010 Conference Record of the Forty Fourth Asilomar Conference on* , vol., no., pp.608-612, 7-10 Nov. 2010

64. **Creusere, C.D.; Siddenki, S.; Hardin, J; Kroger, J**; , "Early investigations into subjective audio quality assessment using brainwave responses," *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on* , vol., no., pp., Nov. 2011.

65. **Castorena, J.E.; Creusere, C;**, "Remote-sensed LIDAR using random sampling and sparse reconstruction," *Proc. International Telemetering Conference*, Las Vegas, NV, October 2011.

66. **Davis, P; Creusere, C.**; "Quantifying the gains of compressive sensing for telemetering application," *Proc. International Telemetering Conference*, Las Vegas, NV, October 2011.

67. **Creusere, C.D.; Kroger, J.; Siddenki, S.R.; Davis, P.; Hardin, J.**; , "Assessment of subjective audio quality from EEG brain responses using time-space-frequency analysis," *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European* , vol., no., pp.2704-2708, 27-31 Aug. 2012.

68. **Castorena, J.; Creusere, C.D.;** , "Compressive sampling of LIDAR: Full-waveforms as signals of finite rate of innovation," *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th*

European , vol., no., pp.984-988, 27-31 Aug. 2012.

69. **Castorena, J.; Creusere, C.D.;** "Random impulsive scan for LIDAR sampling," Proc. IEEE Int. Conf. on Image Proc., October 2012.

70. **Creusere, C. ,Nelson, E., Critz, T., Butcher, E.**; "Analysis of communication interconnectedness in the proximity of near-earth asteroids," *Proc. International Telemetering Conference*, San Diego, CA, October 2012.

71. **Castorena, J.E.; Creusere, C;**, "Remote-sensed LIDAR using random impulsive scans," *Proc. International Telemetering Conference*, San Diego, CA, October 2012.

72. **Castorena, J.; Creusere, C.D.**, "Sub-spot localization for spatial super-resolved LIDAR," *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on* , vol., no., pp.2227,2231, 26-31 May 2013.

73. **Charles D. Creusere ; Juan Castorena**, "A unified framework for 3rd generation lidar pulse processing based on finite rate of innovations," *Proc. SPIE 8858, Wavelets and Sparsity XV*, 88580T (September 26, 2013); doi:10.1117/12.2022447.

74. **Nelson, E., Creusere, C., Critz, T., Butcher, E.**; "Analysis of communication rates in the proximity of near-earth asteroids," *Proc. International Telemetering Conference*, Las Vegas, NV, October 2013.

75. **Castorena, J.E.; Creusere, C;**, "Full-waveform LIDAR recovery at sub-Nyquist rates," *Proc. International Telemetering Conference*, Las Vegas, NV, October 2013.

76. **Davis, P.; Creusere, C.D.; Kroger, J.**, "EEG and the human perception of video quality: Impact of channel selection on discrimination," *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE* , vol., no., pp.9,12, 3-5 Dec. 2013, doi: 10.1109/GlobalSIP.2013.6736798.

77. **Creusere, C.D.; McRae, N.; Davis, P.**, "Sample-based cross-frequency coupling analysis with CFAR detection," *Signals, Systems and Computers, 2014 48th Asilomar Conference on* , vol., no., pp.179,183, 2-5 Nov. 2014; doi: 10.1109/ACSSC.2014.7094423.

78. **Davis, P.; Creusere, C.D.; Kroger, J.**, "Classification of human viewers using high-resolution EEG with SVM," *Signals, Systems and Computers, 2014 48th Asilomar Conference on* , vol., no., pp.184,188, 2-5 Nov. 2014; doi: 10.1109/ACSSC.2014.7094424.

79. **Davis, P.; Creusere, C. D.; Tang, W**., "ASIC implementation of the cross frequency coupling algorithm for EEG signal processing," *2014 14th International Symposium on Integrated Circuits (ISIC)*, pp.248-251, Dec. 2014 doi: 10.1109/ISICIR.2014.7029468

80. **Nelson, Evan; Creusere, Charles D.; Butcher, Eric**, "Determining position around an asteroid using communication relays and trilateration," *Aerospace Conference, 2015 IEEE* , vol., no., pp.1,6, 7-14 March 2015; doi: 10.1109/AERO.2015.7118955.

81. **Davis, Philip; Creusere, Charles D.; Tang, Wei**, "Window length effect on cross frequency coupling in an EEG processing circuit," in *Circuits and Systems (MWSCAS), 2015 IEEE 58th International Midwest Symposium on* , vol., no., pp.1-4, 2-5 Aug. 2015 doi: 10.1109/MWSCAS.2015.7282125.

82. **Davis, Philip; Creusere, Charles D.; Kroger, Jim,** "Subject Identification Based on EEG Response to Video Stimuli," *Proceedings International Conference on Image Processing, 2015 IEEE*, September 2015.

83. **Newtson, Kathy; Creusere, Charles D.**, "Histogram oriented gradients and map seeking circuits pattern recognition with compressed imagery," *Proc. Southwest Symposium on Image Analysis and Interpretation*, Santa Fe, NM, March 2016.

84. **Davis, Philip; Creusere, Charles D.; Kroger, Jim**, " Assessing Cross Frequency Coupling in EEG Collected from Subjects Viewing Video using a Modified Metric ," *Proc. Southwest Symposium on Image Analysis and Interpretation*, Santa Fe, NM, March 2016.

85. **Davis, Philip; Creusere, Charles D.; Kroger, Jim**, "The Effect of Perceptual Video Quality on EEG Power Distribution," *Proceedings International Conference on Image Processing, 2016 IEEE*,

Phoenix, AZ, September 2016.

86. **Creusere, Charles D.; Castorena, J.; Dragulin, I.; Voelz, D.**, "Quantifying the Accuracy of FRI_based LIDAR Waveform Analysis," *Proceedings International Conference on Image Processing, 2016 IEEE*, Phoenix, AZ, September 2016.

87. **Kathy A. Newtson, Charles D. Creusere**, "Compressed imagery detection rate through map seeking circuit, and histogram of oriented gradient pattern recognition", Proc. SPIE 10203, Pattern Recognition and Tracking XXVIII, 102030F (1 May 2017); doi:10.1117/12.2262919; https://doi.org/10.1117/12.2262919

88. **Erandi Wijerathna, Charles D. Creusere, David Voelz, Juan Castorena**, "Polarimetric LIDAR with FRI sampling for target characterization", Proc. SPIE 10407, Polarization Science and Remote Sensing VIII, 104070R (7 September 2017); doi: 10.1117/12.2272587; https://doi.org/10.1117/12.2272587

89. **Andrew J. Phillips, Charles D. Creusere**, "The effects of lossy EEG compression on ERP analysis," *International Telemetering Conference Proceedings*, November 2018, http://hdl.handle.net/10150/631654.

90. **Andrew J. Phillips, Charles D. Creusere**, "The effects of lossy frequency-domain EEG compression on cross-frequency coupling analysis," *International Telemetering Conference Proceedings*, October 2019.

91. **Sergio Lara, Charles D. Creusere,** "Software Defined Radio for Carrier and Symbol Timing Extraction," *International Telemetering Conference Proceedings*, October 2021.

92. **Sergio Lara, Charles D. Creusere,** "LabView for Software Defined Radio Development," *International Telemetering Conference Proceedings*, October 2021.

93. **Ayatelrahman Elsayed, Charles D. Creusere**, "The Feedback Dynamics of Brain-Computer Interfaces in a Distributed Processing Environment," *Proceedings of HICSS-55*, January 2022; http://hdl.handle.net/10125/79821.

94. **Joshua Tellez, Charles D. Creusere,** "Evaluation of JPEG-2000 Image Compression as Applied to Electroencephalography," *International Telemetering Conference Proceedings*, Glendale, AZ, October 2022, https://repository.arizona.edu/handle/10150/666964.

PRESENTATIONS & OTHER PUBLICATIONS

1. **C.D. Creusere**, "Generic Tracking Laboratory Study," *NWC Technical Paper 7018*, Nov. 1989, Naval Weapons Center, China Lake, CA.

2. **C.D. Creusere**," Robust image coding using the embedded zerotree wavelet algorithm," presented at the Digital Compression Conference, Snowbird, UT, April 1996.

3. **C.D. Creusere**, "Creating a compressed and embedded image representation which is robust to transmission errors," presented at the Office of Naval Research Workshop on Error Resilient Compression, San Diego, CA, February, 1996.

4. **C.D. Creusere**, "Data Compression Project Final Report," NAWCWD Technical Paper, TP 8442.

5. **C.D. Creusere**: "Applications of multirate digital signal processing to communications systems," NATO/RTA lecture series 216 on *Applications of M*athematical *Signal Processing Techniques to Mission Systems*, presented in Paris, France; Cologne, Germany; and Monterey, CA, November 1999.

6. **C.D. Creusere**, "Object detection and recognition in compressed video," invited presentation

and white paper for the Motion Imagery Workshop, sponsored by the National Imagery and Mapping Agency (NIMA) and the National Science Foundation (NSF).

7. **C.D. Creusere**, "Application specific compression: digital elevation map browsing, video surveillance, and scalable audio," invited seminar at the University of Arizona, Tucson, AZ, March 2003.

8. **C.D. Creusere**, "Compressing Digital Elevation Maps for Efficient Search and Retrieval," presented at the Office of Naval Research Workshop on Image Processing, Minneapolis, MN, May 2003.

All cases since 2014:

- Technical expert for defense (Evaluated expired PD patent and wrote Dec. demonstrating Dell could not have infringed, case was dropped); Case: Princeton Digital v. Dell, 2014-2015, case dismissed June 2015, **no deposition**.
  -> A company named Princeton Digital was threatening to sue Dell over possible infringement on an expired patent related to JPEG image compression. I did some analysis to show that PD COULD have easily determined prior to patent expiration whether or not Dell actually infringed.

- Technical expert for defense; Noninfringement & IPR (6,597,812), Real-time v. SAP/HP Enterprises, 1/2016-6/2016, **2 depositions (IPR)**. **-> the patents in suit were related to lossless compression technology. Patent in IPR suit dealt with combined run-length coding and dictionary-based coding.**
- Technical expert for defense; IPRs (7,378,992 & 7,415,513), Real-time v. Riverbed (and others), 2/2016-present, **1 deposition**. **-> The patents in suit were related to lossless compression technology with the specific being to provide content dependent and independent compression**
- Technical expert for defense; IPRs (8,643,530, 7,378,992, 7,161,506, & 9,054,728), Real-time v.
- Dell (and others), 2/2016-7/2018, **2 depositions**. -> **All of the patents in suit were related to lossless data compression. One set dealt with "accelerated data storage and retrieval" while the other was related to adaptive content dependent/independent compression**
- Technical expert for defense; Noninfringement, Real-time v. HP Enterprises, 4/2016-10/2018, **not deposed, *settled before deposition*. -> Dealt with noninfringement for patents listed for the IPRs above.**
- Technical expert for defense; IPR (7,358,867, 7,161,506, & 9,054,728), Real-time v. Teradata
- 11/2016-8/2018, **1 deposition**. **-> Patents dealt with lossless compression. Specifically, patents claimed that different encoding schemes would be selected for each of a plurality of data blocks based on some criterion. Specific to one of these patents, the encoder could also choose NOT to encode a block if that was desirable.**
- Technical expert for defense; IPRs (8,643,513 & 7,378,992), Real-time v. Veritas 12/2016-2018, **no additional deposition -> Duplicate of Dell IPR above**
- Technical expert for defense; IPRs (7,151,916; 7,075,917; 6,304,612) , Uniloc LLC v. Apple, 2018-2019, **not deposed**. **-> Patents dealt with aspects of communications systems, including tail-biting convolutional coding and ARQ systems for error correction**
- Technical expert for defense; Noninfringement, Realtime Adaptive Systems v. YouTube/Google, 2018-2019, **1 deposition, *settled before trial***.
- Technical expert for defense; IPR (7,558,730), AVRS v. Apple, 2019-2020, **2 depositions**. **-> dealt with an internet-based voice transcription service.**
- Technical expert for defense; IPR (10,129,590), Maxell v. Apple, 2019-2020, Oct. 2020-2021. -> **Dealt with video compression. Specifically, a methodology for capturing video from a device and transmitting it wirelessly to at least two other devices (one through the internet) where one device has a higher priority than the other. *Settled before deposition*.**
- Technical expert for defense; U.S. ITC case; Nokia v. Lenovo, Oct. 2020-March 2021 -> **Patents related to H.264 video compression**. Hired as a testifying expert witness but ultimately have provided primarily technical analysis of opposing expert witness report. ***No deposition.***

- Technical expert for defense; IPR (10,176,848) , Maxell v. Apple, 2019-2020, Dec. 2020-2021. -> **Deals with video compression. Specifically, a methodology for identifying a person of interest in recorded digital video.** ***Settled before deposition*****.**
- Technical expert for defense; Claims construction, Gesture Tech v. Apple/Lenovo/Motorola, 11/2021-12/2021, 8/2023. ***No depositions***.
- Technical expert for plaintiff; USAA v. PNC Bank; October 2021-January 2023; Expert Arguing against invalidity in two U.S. District Court cases (2:20-cv-00319-JRG and 2:21-cv-00246-JRG) and 4 IPRs (IPR2022-00075, IPR2022-00076, IPR2021-01073, and IPR2021-01070). **All patents dealt with mobile remote check deposit.** ***Deposed 4 times and attended both trials but did not testify***.
- Technical expert for the defense; US ITC case; VideoLabs, Inc. v. HP Inc; August 2023 scheduled; ***Patents invalidated by IPRs prior to start of ITC case.***
- Technical expert for defense; Federal District Court; Gesture Partners v. Lenovo/Motorola, December 2023-present. Wrote invalidity and noninfringement reports (source code and device analysis), Deposed April 2024. USP 8,878,949 dealt with optical-based gesture recognition. **Summary judgement of noninfringement, September 2024**.
- Technical expert for defense; IPRs; Nokia v. Amazon; USP 10,536,714, USP 8,077,991; 2024-present; **Depositions pending institution**.
- Technical expert for defense; IPRs; Maxell v. Samsung; USP 11,445,241; 2024-present; **Depositions pending institution**.
- Technical expert for defense; Ex Parte Reexam; USP 6,389,473; BSD Crown LTD. v. Amazon; July-August 2024.
- Technical expert for the defense; Federal District Court/ITC; Nokia v. HP Inc; 2024-present; **Case pending**.

APPENDIX 2: MATERIALS CONSIDERED IN THE PREPARATION OF THIS DECLARATION

Exhibit No.	Description
1001	U.S. Patent No. 9,571,833 to Bici, et al. ("the '833 patent")
1002	Certified Prosecution File History for the '833 patent ("Prosecution History")
1004	U.S. Patent Application Publication No. 2011/0194609 to Rusert et al. ("Rusert")
1005	U.S. Patent Application Publication No. 2013/0077691 to Zheng et al. ("Zheng")
1006	U.S. Patent Application Publication No. 2014/0092981 to Lin et al. ("Lin")
1007	Nakamura et al., "Unification of derivation process for merge mode and MVP," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 6th Meeting, Torino, Italy, Jul. 14-22, 2011, Document JCTVC-F419 ("Nakamura Document")
1008	Nakamura et al., "WD description of JCTVC-F419 Proposal 1" for "Unification of derivation process for merge mode and MVP," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 6th Meeting, Torino, Italy, Jul. 14-22, 2011, Document JCTVC-F419 ("Nakamura WD Description")
1009	Nakamura et al., "Unification of derivation process for merge mode and MVP" Presentation, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 6th Meeting, Torino, Italy, Jul. 14-22, 2011, Document JCTVC-F419 ("Nakamura Presentation")
1010	Bross et al., "WD4: Working Draft 4 of High-Efficiency Video Coding," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 6th Meeting: Torino, IT, Jul. 14-22, 2011, Document JCTVC-F803 ("WD4")
1011	U.S. Patent Application Publication No. 2011/0249721 to Karczewicz et al ("Karczewicz")
1012	U.S. Provisional Application 61/301,649 ("Rusert Provisional")
1013	INTENTIONALLY LEFT BLANK
1014	Declaration of Vivienne Sze ("Sze Declaration")

Exhibit No.	Description
1015	Prosecution File History for U.S. Patent No. 9,571,714 ("'714 Prosecution File History")
1016	Prosecution File History for U.S. Patent No. 9,743,105 ("'105 Prosecution File History")
1017	European Prosecution File History for 12845839 ("European Prosecution History")
1018	U.S. Patent Application Publication No. 2012/0128067 to Liu ("Liu")
1019	Gary J. Sullivan, Recent Developments in Standardization of High Efficiency Video Coding (HEVC) ("Sullivan")
1020	Frank Bossen, HEVC Complexity and Implementation Analysis ("Bossen")
1021	Jian-Wei Chen et al, Introduction to H.264 advanced video coding. In: Proceedings of the 2006 Asia and South Pacific Design Automation Conference vol. 2006, pp. 736–741 (2006)
1022	Won, Kwanghyun et al, Motion vector coding using decoder-side estimation of motion vector. In 2009 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB). IEEE, 2009.
1023	Gary J. Sullivan et al, The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions, SPIE Conference on Applications of Digital Image Processing XXVII, 2004.
1024	Gweon, Ryeong-Hee, Non-Fixed Quantization Considering Entropy Encoding in HEVC, Journal of Broadcast Engineering. The Korean Institute of Broadcast and Media Engineers, 2011
1025	Hong, Chang-Yi, New Merge Mode Decision in High Efficiency Video Coding (HEVC), 2014 International Computer Science and Engineering Conference (ICSEC), IEEE (2014).
1026	JCT-VC - Joint Collaborative Team on Video Coding (https://www.itu.int/en/ITU-T/studygroups/2013-2016/16/Pages/video/jctvc.aspx)
1027	Terms of Reference of the Joint Collaborative Team on Video Coding Standard Development (https://www.itu.int/dms_pub/itu-t/oth/46/01/T46010000010001PDFE.pdf)

Exhibit No.	Description
1028	Sullivan et al., "Meeting report of the sixth meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Torino, IT, 14–22 July 2011," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 6th Meeting, Torino, Italy, Jul. 14-22, 2011, Document JCTVC-F800
1029	"All Meetings" web page (from the JCT-VC site) (http://phenix.int-evry.fr/jct/doc_end_user/all_meeting.php)
1030	"Torino Meeting - Document Register" web page (from the JCT-VC site) (http://phenix.int-evry.fr/jct/doc_end_user/current_meeting.php?id_meeting=149&search_id_group=1&search_sub_group=1)
1031	A portion of search screen for "merge mode" on the "Torino Meeting - Document Register" web page
1032	A portion of search screen for "MVP" on the "Torino Meeting - Document Register" web page
1033	Search result page for "merge mode" originated from the "Torino Meeting - Document Register" web page
1034	Search result page for "MVP" originated from the "Torino Meeting - Document Register" web page
1035	Search result page for "WD4" originated from the "Torino Meeting - Document Register" web page
1036	Search result page for "high-efficiency video coding" originated from the "Torino Meeting - Document Register" web page
1037	Search result page for "DRAFT" originated from the "Torino Meeting - Document Register" web page
1038	"Preview document JCTVC-F419 for Torino meeting" web page (http://phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=2894)
1039	"Preview document JCTVC-F803 for Torino meeting" web page (http://phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=3286)
1040	Sullivan et al., "Meeting report of the third meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Guangzhou, CN, 7–15 October, 2010," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 3rd Meeting, Guangzhou China, Oct. 7-15, 2010, Document JCTVC-C400

Exhibit No.	Description
1041	"All Meetings" web page (from the JCT-VC distribution website) (http://phenix.int-evry.fr/jct/doc_end_user/all_meeting.php)
1042	"Torino Meeting - Document Register" web page (from the JCT-VC distribution website) (http://phenix.int-evry.fr/jct/doc_end_user/current_meeting.php?id_meeting=149&search_id_group=1&search_sub_group=1)
1043	Search result page for "merge mode" originated from the "Torino Meeting - Document Register" web page (from the JCT-VC distribution website)
1044	Search result page for "MVP" originated from the "Torino Meeting - Document Register" web page (from the JCT-VC distribution website)
1045	Search result page for "WD4" originated from the "Torino Meeting - Document Register" web page (from the JCT-VC distribution website)
1046	Search result page for "WD" originated from the "Torino Meeting - Document Register" web page (from the JCT-VC distribution website)
1047	"Preview document JCTVC-F419 for Torino meeting" web page (http://phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=2894)
1048	"Preview document JCTVC-F803 for Torino meeting" web page (http://phenix.int-evry.fr/jct/doc_end_user/current_document.php?id=3286)
1049	Search result page for "Nakamura" originated from the "Torino Meeting - Document Register" web page (from the JCT-VC distribution website)
1050	Declaration of Clifford Reader
1051	Machine translation of Gweon, Ryeong-Hee, Non-Fixed Quantization Considering Entropy Encoding in HEVC, Journal of Broadcast Engineering. The Korean Institute of Broadcast and Media Engineers, 2011
1052	Prosecution File History for U.S. Patent No. 10,237,574 ("'574 Prosecution File History")
1053	Prosecution File History for U.S. Patent No. 11,570,467 ("'467 Prosecution File History")

Exhibit No.	Description
1054	U.S. Provisional Application 61/499,112 ("Zheng '112 Provisional")
1055	U.S. Provisional Application 61/543,043 ("Zheng '043 Provisional")
1056	U.S. Provisional Application 61/543,059 ("Zheng '059 Provisional")
1057	U.S. Provisional Application 61/556,761 ("Zheng '761 Provisional")
1058	U.S. Provisional Application 61/562,387 ("Zheng '387 Provisional")
1059	U.S. Provisional Application 61/562,953 ("Zheng '953 Provisional")
1060	Oudin, S., et al., Block Merging for Quadtree-Based Video Coding. In 2011 IEEE International Conference on Multimedia and Expo. IEEE, July 11-15, 2011.
1061	U.S. Patent No. 9,313,494 to Zheng et al. ("Zheng-Issued")
1062	Prosecution File History for U.S. Patent No. 9,313,494 ("Zheng-Issued File History")

APPENDIX 3: CHALLENGED CLAIMS

9. A method comprising:

[9a] receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;

[9b] determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;

[9c] selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;

[9d] determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;

[9e] comparing motion information of the first spatial motion vector prediction candidate with motion information of another spatial motion vector prediction candidate of the [sub]set of spatial motion vector prediction candidates

without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

[9f] if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and

[9g] selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.

10. The method according to claim 9 comprising comparing motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.

11. The method according to claim 9 comprising examining whether the received encoded block of pixels is divided into a first prediction unit and a second

prediction unit; and if so, excluding the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.

12. The method according to claim 9 further comprising

[12a] determining a maximum number of spatial motion vector prediction candidates to be included in a merge list; and

[12b] limiting the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number.

13. The method according to claim 12 comprising:

[13a] examining, if the number of spatial motion vector prediction candidates in the merge list smaller than the maximum number;

[13b] if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[13c] if so, performing at least one of the following:

[13d] for a potential spatial motion vector prediction candidate on a left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[13e] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[13f] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;

[13g] for a potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[13h] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;

[13i] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;

[13j] for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate

from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[13k] for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and

[13l] for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, excluding the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[13m] all the other potential spatial motion vector prediction candidates have been included in the merge list;

[13n] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[13o] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.

14. The method according to claim 9 comprising selecting one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.

16. An apparatus comprising a processor and a memory including computer program code, the memory and the computer program code configured to, with the processor, cause the apparatus to:

[16a] receive an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;

[16b] determine a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;

[16c] select a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the

merge list is constructed based on the motion information of the spatial motion vector prediction candidates;

[16d] determine a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;

[16e] compare motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

[16f] exclude the first spatial motion vector prediction candidate from the merge list, if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other; and

[16g] select a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector

prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.

18. A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprise instructions for use by an encoder, said program code comprises instructions for:

[18a] receiving an encoded block of pixels including a prediction unit and information identifying a respective spatial motion vector prediction candidate from a merge list constructed by an encoder;

[18b] determining a set of spatial motion vector prediction candidates for the encoded block of pixels; the spatial motion vector prediction candidates being provided with motion information;

[18c] selecting a first spatial motion vector prediction candidate from the set of spatial motion vector prediction candidates as a potential spatial motion vector prediction candidate to be included in a merge list for the prediction unit, where the merge list is constructed based on the motion information of the spatial motion vector prediction candidates;

[18d] determining a subset of spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;

[18e] comparing motion information of the first spatial motion vector prediction candidate with motion information of the spatial motion vector prediction candidate in the determined subset of spatial motion vector prediction candidates without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates, wherein comparing comprises performing an equivalence check or comparing a difference in motion information to a threshold or other similarity metric;

[18f] if at least one of the comparisons indicates that the motion vector information of the spatial motion vector prediction candidates correspond with each other, excluding the first spatial motion vector prediction candidate from the merge list; and

[18g] selecting a spatial motion vector prediction candidate from the merge list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the merge list using the information that was received identifying a respective spatial motion vector prediction candidate.

26. The apparatus according to claim 16 wherein the apparatus is further caused to compare motion information of the potential spatial motion vector prediction candidate with motion information of at most one other spatial motion vector prediction candidate of the set of spatial motion vector prediction candidates.

27. The apparatus according to claim 16 wherein the apparatus is further caused to examine whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit; and if so, exclude the potential spatial motion vector prediction candidate from the merge list if the prediction unit is the second prediction unit.

28. The apparatus according to claim 16 wherein the apparatus is further caused to:

[28a] determine a maximum number of spatial motion vector prediction candidates to be included in a merge list; and

[28b] limit the number of spatial motion vector prediction candidates in the merge list smaller or equal to the maximum number.

29. The apparatus according to claim 28 wherein the apparatus is further caused to:

[29a] examine if the number of spatial motion vector prediction candidates in the merge list smaller than the maximum number;

[29b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[29c] if so, perform at least one of the following:

[29d] for a potential spatial motion vector prediction candidate on a left side of the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29e] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[29f] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and if the prediction unit is the second prediction unit, and the potential spatial motion vector prediction candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;

[29g] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29h] the received encoded block of pixels is horizontally divided into a first prediction unit and a second prediction unit, and the prediction unit is the second prediction unit;

[29i] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit;

[29j] for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[29k] for a potential spatial motion vector prediction candidate, which is below the potential spatial motion vector prediction candidate on the left side of the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit; and

[29l] for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, exclude the potential spatial motion vector prediction candidate from the merge list if any of the following conditions are fulfilled:

[29m] all the other potential spatial motion vector prediction candidates have been included in the merge list;

[29n] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[29o] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate on the left side of the prediction unit.

30. The apparatus according to claim 16 wherein the apparatus is further caused to select one motion vector prediction candidate from the merge list to represent a motion vector prediction for the received encoded block of pixels.

APPENDIX 4: UNDERSTANDING OF THE LAW

I have applied the following legal principles provided to me by counsel in arriving at the opinions set forth in this report.

Legal Standard for Prior Art

I am not an attorney. I have been informed by attorneys of the relevant legal principles and have applied them to arrive at the opinions set forth in this declaration.

I understand that the petitioner for inter partes review may request the cancelation of one or more claims of a patent based on grounds available under 35 U.S.C. § 102 and 35 U.S.C. § 103 using prior art that consists of patents and printed publications.

Anticipation and Prior Art

I understand that § 102 specifies when a challenged claim is invalid for lacking novelty over the prior art, and that this concept is also known as "anticipation." I understand that a prior art reference anticipates a challenged claim, and thus renders it invalid by anticipation, if all elements of the challenged claim are disclosed in the prior art reference. I understand the disclosure in the prior art reference can be either explicit or inherent, meaning it is necessarily present or implied. I understand that the prior art reference does not have to use the same words as the challenged claim, but all of the requirements of the claim

must be disclosed so that a person of ordinary skill in the art could make and use the claimed subject-matter.

In addition, I understand that § 102 also defines what is available for use as a prior art reference to a challenged claim.

Under § 102(a), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country before the challenged claim's date of invention.

Under § 102(b), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country more than one year prior to the challenged patent's filing date.

Under § 102(e), a challenged claim is anticipated if it was described in a published patent application that was filed by another in the United States before the challenged claim's date of invention, or was described in a patent granted to another that was filed in the United States before the challenged claim's date of invention.

I understand that a challenged claim's date of invention is presumed to be the challenged patent's filing date. I also understand that the patent owner may establish an earlier invention date and "swear behind" prior art defined by § 102(a) or § 102(e) by proving (with corroborated evidence) the actual date on which the

named inventors conceived of the subject matter of the challenged claim and proving that the inventors were diligent in reducing the subject matter to practice.

I understand that the filing date of a patent is generally the filing date of the application filed in the United States that issued as the patent. However, I understand that a patent may be granted an earlier effective filing date if the patent owner properly claimed priority to an earlier patent application.

I understand that when a challenged claim covers several structures, either generically or as alternatives, the claim is deemed anticipated if any of the structures within the scope of the claim is found in the prior art reference.

I understand that when a challenged claim requires selection of an element from a list of alternatives, the prior art teaches the element if one of the alternatives is taught by the prior art.

Legal Standard for Obviousness

I understand that even if a challenged claim is not anticipated, it is still invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter would have been obvious to a person of ordinary skill in the pertinent art at the time the alleged invention.

I understand that obviousness must be determined with respect to the challenged claim as a whole.

I understand that one cannot rely on hindsight in deciding whether a claim is obvious.

I also understand that an obviousness analysis includes the consideration of factors such as (1) the scope and content of the prior art, (2) the differences between the prior art and the challenged claim, (3) the level of ordinary skill in the pertinent art, and (4) "secondary" or "objective" evidence of non-obviousness.

Secondary or objective evidence of non-obviousness includes evidence of: (1) a long felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others. However, I understand that there must be a relationship between any secondary evidence of non-obviousness and the claimed invention.

I understand that a challenged claim can be invalid for obviousness over a combination of prior art references if a reason existed (at the time of the alleged invention) that would have prompted a person of ordinary skill in the art to combine elements of the prior art in the manner required by the challenged claim. I understand that this requirement is also referred to as a "motivation to combine,"

"suggestion to combine," or "reason to combine," and that there are several rationales that meet this requirement.

I understand that the prior art references themselves may provide a motivation to combine, but other times simple common sense can link two or more prior art references. I further understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may come from market forces.

I understand obviousness to include, for instance, scenarios where known techniques are simply applied to other devices, systems, or processes to improve them in an expected or known way. I also understand that practical and common-sense considerations should be applied in a proper obviousness analysis. For instance, familiar items may have obvious uses beyond their primary purposes.

I understand that the combination of familiar elements according to known methods is obvious when it yields predictable results. For instance, obviousness bars patentability of a predictable variation of a technique even if the technique originated in another field of endeavor. This is because design incentives and other market forces can prompt variations of it, and predictable variations are not the product of innovation, but rather ordinary skill and common sense.

I understand that a particular combination may be obvious if it was obvious to try the combination. For example, when there is a design need or market

pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. This would result in something obvious because the result is the product not of innovation but of ordinary skill and common sense. However, I understand that it may not be obvious to try a combination when it involves unpredictable technologies.

It is further my understanding that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

It is my understanding that the Manual of Patent Examining Procedure §2143 sets forth the following as exemplary rationales that support a conclusion of obviousness:

- 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 technique 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;
- Choosing 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;
- Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

A person of ordinary skill in the art looking to overcome a problem will often use the teachings of multiple publications together like pieces of a puzzle, even though the prior art does not necessarily fit perfectly together. Therefore, I understand that references for obviousness need not fit perfectly together like puzzle pieces. Instead, I understand that obviousness analysis takes into account inferences, creative steps, common sense, and practical logic and applications that a person of ordinary skill in the art would employ under the circumstances.

I understand that a claim can be obvious in light of a single reference, if the elements of the challenged claim that are not explicitly or inherently disclosed in the reference can be supplied by the common sense of one of skill in the art.

I understand that obviousness also bars the patentability of applying known or obvious design choices to the prior art. One cannot patent merely substituting one prior art element for another if the substitution can be made with predictable results. Likewise, combining prior art techniques that are interoperable with respect to one another is generally obvious and not patentable.

In order for a claim to be found invalid based upon a modification or combination of the prior art, there must be reasonable expectation that a person of ordinary skill would have successfully modified or combined the prior art to arrive at the claimed arrangement. This does not mean that it must be certain that a person of ordinary skill would have been successful – the law only requires that the person of ordinary skill in the art would have perceived a reasonable expectation of success in modifying or combining the prior art to arrive at the claimed invention.

In sum, my understanding is that obviousness invalidates claims that merely recite combinations of, or obvious variations of, prior art teachings using understanding and knowledge of one of skill in the art at the time and motivated by the general problem facing the inventor at the time. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor

at the time of the invention, can provide a reason for combining the elements of or attempting obvious variations on prior art references in the claimed manner.

Legal Standard for Claim Construction

I understand that before any invalidity analysis can be properly performed, the scope and meaning of the challenged claims must be determined by claim construction.

I understand that a patent may include two types of claims, independent claims and dependent claims. I understand that an independent claim stands alone and includes only the limitations it recites. I understand that a dependent claim depends from an independent claim or another dependent claim. I understand that a dependent claim includes all the limitations that it recites in addition to the limitations recited in the claim (or claims) from which it depends.

In comparing the challenged claims to the prior art, I have carefully considered the patent and its file history in light of the understanding of a person of skill at the time of the alleged invention.

I understand that to determine how a person of ordinary skill would have understood a claim term, one should look to sources available at the time of the alleged invention that show what a person of skill in the art would have understood disputed claim language to mean. It is my understanding that this may include what is called "intrinsic" evidence as well as "extrinsic" evidence.

I understand that, in construing a claim term, one should primarily rely on intrinsic patent evidence, which includes the words of the claims themselves, the remainder of the patent specification, and the prosecution history. I understand that extrinsic evidence, which is evidence external to the patent and the prosecution history, may also be useful in interpreting patent claims when the intrinsic evidence itself is insufficient. I understand that extrinsic evidence may include principles, concepts, terms, and other resources available to those of skill in the art at the time of the invention.

I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else or something more specific. I understand that to determine whether a term has special meaning, the claims, the patent specification, and the prosecution history are particularly important, and may show that the inventor gave a term a particular definition or intentionally disclaimed, disavowed, or surrendered claim scope.

I understand that the claims of a patent define the scope of the rights conferred by the patent. I understand that because the claims point out and distinctly claim the subject matter which the inventors regard as their invention, claim construction analysis must begin with and is focused on the claim language itself. I understand that the context of the term within the claim as well as other claims of the patent can inform the meaning of a claim term. For example, because

claim terms are normally used consistently throughout the patent, how a term is used in one claim can often inform the meaning of the same term in other claims. Differences among claims or claim terms can also be a useful guide in understanding the meaning of particular claim terms.

I understand that a claim term should be construed not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the entire specification. I understand that because the specification is a primary basis for construing the claims, a correct construction must align with the specification.

I understand that the prosecution history of the patent as well as art incorporated by reference or otherwise cited during the prosecution history are also highly relevant in construing claim terms. For instance, art cited by or incorporated by reference may indicate how the inventor and others of skill in the art at the time of the invention understood certain terms and concepts. Additionally, the prosecution history may show that the inventors disclaimed or disavowed claim scope, or further explained the meaning of a claim term.

With regard to extrinsic evidence, I understand that all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may indicate how one of skill in the art used or understood the claim

terms. However, I understand that extrinsic evidence is considered to be less reliable than intrinsic evidence, and for that reason is generally given less weight than intrinsic evidence.

I understand that in general, a term or phrase found in the introductory words or preamble of the claim, should be construed as a limitation if it recites essential structure or steps, or is necessary to give meaning to the claim. For instance, I understand preamble language may limit claim scope: (i) if dependence on a preamble phrase for antecedent basis indicates a reliance on both the preamble and claim body to define the claimed invention; (ii) if reference to the preamble is necessary to understand limitations or terms in the claim body; or (iii) if the preamble recites additional structure or steps that the specification identifies as important.

On the other hand, I understand that a preamble term or phrase is not limiting where a challenged claim defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention. I understand that to make this determination, one should review the entire patent to gain an understanding of what the inventors claim they invented and intended to encompass in the claims.

I understand that 35 U.S.C. § 112 ¶ 6 created an exception to the general rule of claim construction called a "means plus function" limitation. These types of

terms and limitations should be interpreted to cover only the corresponding structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if (a) the claim limitation uses the phrase "means for"; (b) the "means for" is modified by functional language; and (c) the phrase "means for" is not modified by sufficient structure for achieving the specified function.

I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the differences between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I further understand that a structural equivalent must have been available at the time of the issuance of the claim.