IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of:	Mehmet Oguz Bici et al.	
U.S. Patent No.:	10,536,714	Attorney Docket No.: 54587-0016IP1
Issue Date:	January 14, 2020	
Appl. Serial No.:	16/356,733	
Filing Date:	March 18, 2019	
Title:	METHOD FOR CODING AND AN APPARATUS	

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

PETITION FOR *INTER PARTES* REVIEW OF UNITED STATES PATENT NO. 10,536,714 PURSUANT TO 35 U.S.C. §§ 311–319, 37 C.F.R. § 42

TABLE OF CONTENTS

EXHIBITS

Exhibit No.	Description
ASUS-1001	U.S. Patent No. 10,536,714 to Bici, et al. (“the ’714 patent”)
ASUS-1002	Prosecution File History for the ’714 patent (“Prosecution History”)
ASUS-1003	Declaration of Joseph Havlicek (“Havlicek Declaration”)
ASUS-1004	U.S. Patent Application Publication No. 2011/0194609 to Rusert et al (“Rusert”)
ASUS-1005	U.S. Patent Application Publication No. 2011/0249721 to Karczewicz et al (“Karczewicz”)
ASUS-1006	U.S. Patent Application Publication No. 2014/0092981 to Lin et al (“Lin”)
ASUS-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”)
ASUS-1008-1011	[RESERVED]
ASUS-1012	U.S. Provisional Application 61/301,649 (“Rusert Provisional”)
ASUS-1013	U.S. Provisional Application 61/500,903 (“Lin Provisional”)
ASUS-1014	[RESERVED]
ASUS-1015	Prosecution File History for U.S. Patent No. 9,571,833 (“’833 Prosecution History”)
ASUS-1016	[RESERVED]
ASUS-1017	European Prosecution File History for 12845839 (“European Prosecution History”)
ASUS-1018	U.S. Patent Application Publication No. 2012/0128067 to Liu (“Liu”)
ASUS-1019	Gary J. Sullivan, Recent Developments in Standardization of High Efficiency Video Coding (HEVC) (“Sullivan”)
ASUS-1020	Frank Bossen, HEVC Complexity and Implementation Analysis (“Bossen”)
ASUS-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)

ASUS-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
ASUS-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
ASUS-1024-1051	[RESERVED]
ASUS-1052	Institution Decision IPR2024-00604, Paper 11, November 7, 2024
ASUS-1053	Institution Decision IPR2024-00605, Paper 11, November 7, 2024

LISTING OF CLAIMS

[1pre] A method comprising:

[1a] selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for a block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates and is utilized to identify motion vector prediction candidates of which one spatial motion vector prediction candidate from the motion vector prediction list is signaled as the motion information for the prediction unit;

[1b] determining a subset of spatial motion vector prediction candidates based on a location of the block associated with the first spatial motion vector prediction candidate;

[1c] 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 pair from the set of spatial motion vector prediction candidates;

[1d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and

[1e] causing information identifying the one spatial motion vector prediction candidate from the motion vector prediction list to be transmitted to a decoder or to be stored.

2. The method according to claim 1 further comprising selecting spatial motion vector prediction candidates from the set of spatial motion vector prediction candidates as the potential spatial motion vector prediction candidate in a predetermined order.

3. The method according to claim 1, further 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.

4. The method according to claim 1 further comprising examining whether the 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 motion vector prediction list if the prediction unit is the second prediction unit.

[5pre] The method according to claim 1, further comprising

[5a] determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and

[5b] limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.

[6pre] The method according to claim 5 comprising:

[6a] examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;

[6b] if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[6c] if so, performing at least one of the following:

[6d] 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 motion vector prediction list if any of the following conditions are fulfilled:

[6e] the received block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[6f] the received 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;

[6g] for a potential spatial motion vector prediction candidate above the prediction unit, excluding the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[6h] the received 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;

[6i] 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;

[6j] 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 motion vector prediction 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;

[6k] 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 motion vector prediction 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;

[6l] for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, excluding the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[6m] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;

[6n] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[6o] 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.

7. The method according to claim 1 further comprising including a

temporal motion prediction candidate into the motion vector prediction list.

8. The method according to claim 1 further comprising selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the block of pixels.

[9pre] A method comprising:

[9a] selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;

[9b] 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;

[9c] 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;

[9d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and

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

10. The method according to claim 9 further 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 further 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 motion vector prediction list if the prediction unit is the second prediction unit.

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

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

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

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

[13a] examining, if the number of spatial motion vector prediction candidates in the motion vector prediction 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 motion vector prediction 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 motion vector prediction 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 motion vector prediction 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 motion vector prediction 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 motion vector prediction list if any of the following conditions are fulfilled:

[13m] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction 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 further comprising selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the encoded block of pixels.

[15pre] 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:

[15a] select a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for a block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates and is utilized to identify motion vector prediction candidates of which one spatial motion vector prediction candidate from the motion vector prediction list is signaled as the motion information for the prediction unit;

[15b] 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;

[15c] 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;

[15d] determine to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on comparison of the motion information of the first spatial motion vector candidate with motion information of the spatial motion vector prediction candidate; and

[15e] cause information identifying the one spatial motion vector prediction candidate from the motion vector prediction list to be transmitted to a decoder or to be stored.

16. The apparatus according to claim 15 wherein the apparatus is further caused to select spatial motion vector prediction candidates from the set of spatial motion vector prediction candidates as the potential spatial motion vector prediction candidate in a predetermined order.

17. The apparatus according to claim 15, 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.

18. The apparatus according to claim 15 wherein the apparatus is

further caused to examine whether the 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 motion vector prediction list if the prediction unit is the second prediction unit.

19. The apparatus according to claim 15, wherein the apparatus is further caused to:

determine a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and

limit the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.

[20pre] The apparatus according to claim 19 wherein the apparatus is further caused to:

[20a] examine, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;

[20b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[20c] if so, perform at least one of the following:

[20d] 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 motion vector prediction list if any of the following conditions are fulfilled:

[20e] the received block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[20f] the received 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;

[20g] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[20h] the received 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;

[20i] 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;

[20j] 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 motion vector prediction 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;

[20k] 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 motion vector prediction 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;

[20l] for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, exclude the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[20m] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;

[20n] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction

candidate above the prediction unit;

[20o] 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.

21. The apparatus according to claim 15 wherein the apparatus is further caused to include a temporal motion prediction candidate into the motion vector prediction list.

22. The apparatus according to claim 15 wherein the apparatus is further caused to select one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the block of pixels.

[23pre] 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:

[23a] select a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;

[23b] 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;

[23c] 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;

[23d] determine to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on comparison of the motion information of the first spatial motion vector candidate with motion information of the spatial motion vector prediction candidate; and

[23e] select a spatial motion vector prediction candidate from the motion vector prediction list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the motion vector prediction list using information that was received identifying a respective spatial motion vector prediction candidate from the motion vector prediction list constructed by an encoder.

24. The apparatus according to claim 23 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.

25. The apparatus according to claim 23 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 motion vector prediction list if the prediction unit is the second prediction unit.

[26pre] The apparatus according to claim 23 wherein the apparatus is further caused to:

[26a] determine a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and

[26b] limit the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.

[27pre] The apparatus according to claim 26 wherein the apparatus is further caused to:

[27a] examine if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;

[27b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;

[27c] if so, perform at least one of the following:

[27d] 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 motion vector prediction list if any of the following conditions are fulfilled:

[27e] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;

[27f] 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;

[27g] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[27h] 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;

[27i] 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;

[27j] 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 motion vector prediction 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;

[27k] 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 motion vector prediction 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

[27l] for a potential spatial motion vector prediction candidate cornerwise neighbouring the prediction unit, exclude the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:

[27m] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;

[27n] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;

[27o] 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.

28. The apparatus according to claim 23 wherein the apparatus is further caused to select one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the received encoded block of pixels.

[29pre] A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprising instructions for:

[29a] selecting a first spatial motion vector prediction candidate from a

set of spatial motion vector prediction candidates for a block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates and is utilized to identify motion vector prediction candidates of which one spatial motion vector prediction candidate from the motion vector prediction list is signaled as the motion information for the prediction unit;

[29b] 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;

[29c] 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;

[29d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and

[29e] causing information identifying the one spatial motion vector

prediction candidate from the motion vector prediction list to be transmitted to a decoder or to be stored.

[30pre] A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprising instructions for:

[30a] selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;

[30b] 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;

[30c] 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;

[30d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and

[30e] selecting a spatial motion vector prediction candidate from the motion vector prediction list for use in decoding the encoded block of pixels, wherein the spatial motion vector prediction candidate is selected from the motion vector prediction list using information that was received identifying a respective spatial motion vector prediction candidate from the motion vector prediction list constructed by an encoder.

The ’714 patent purports to improve on de-duplicating lists of motion vector prediction candidates by comparing subsets of candidates rather than every possible pair. However, its alleged improvement was already known for the predecessor standard H.264 (Grounds 1-2).

I. BACKGROUND

A. Motion Prediction Basics

Since the 1990s, video-coding standards have applied motion prediction to blocks (e.g., 8x8 pixel squares) in a video frame. ASUS-1023, 000004, Figs. 1-2. This technique used motion vectors to describe the movement of blocks between frames. ASUS-1021, 000002.[1]

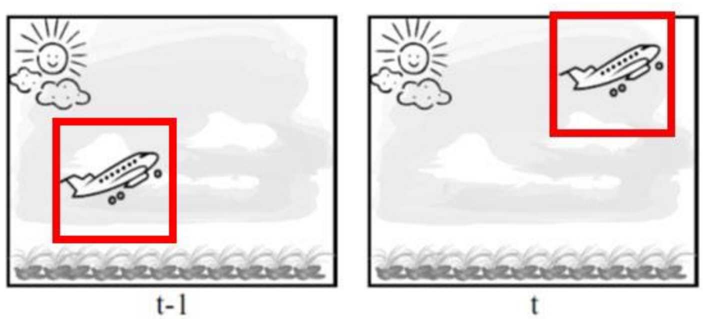
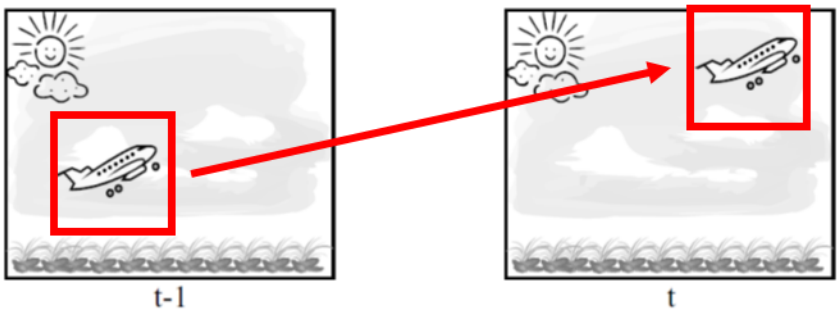


Fig. 3. Successive video frames

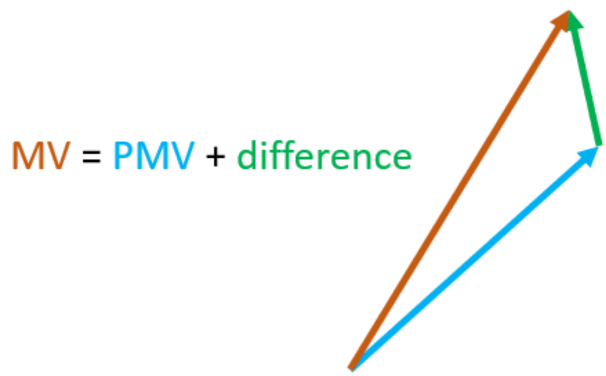
Instead of transmitting an entire block of pixels, an encoder signaled a motion vector, which the decoder then used to locate the current block in a previously-coded frame (e.g., showing the airplane). The decoder then copied the

[1] All emphasis/annotations added unless stated otherwise.

pixels to the current block, thus eliminating the need for those pixels to be resent by the encoder. ASUS-1021, 000002-4; ASUS-1022, 000001-3; ASUS-1003, ¶¶CA3-8.

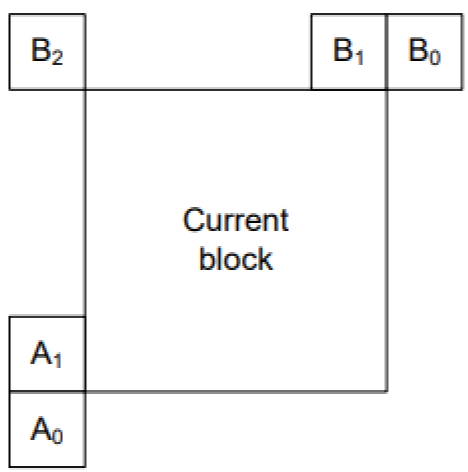


Many video-coding standards, including H.264, further reduced the size of video data by predicting motion vectors for a current block using motion vectors for prior blocks, and sending the difference between the predicted motion vector (“PMV”) or motion vector predictor (“MVP”) and the actual motion vector. This difference was typically much smaller than the full motion vector and thus required fewer bits. Since the encoder and decoder independently calculated the PMV, only the difference vector was transmitted, and the decoder calculated the actual motion vector by adding the PMV to the difference vector. ASUS-1022, 000001-2.



ASUS-1003, ¶CA9.

By 2010, neighboring blocks in the same frame were commonly used to find PMV candidates. Candidates from the same frame were called *spatial* motion vector prediction candidates. ASUS-1022, 000001-3. This was known for H.264. ASUS-1004, ¶¶2-3. Early H.265 drafts also obtained PMV candidates from neighboring blocks, which tended to share the same motion and were thus strong candidates for the current block. Neighboring blocks were labelled based on their position relative to the current block. ASUS-1007, 5:



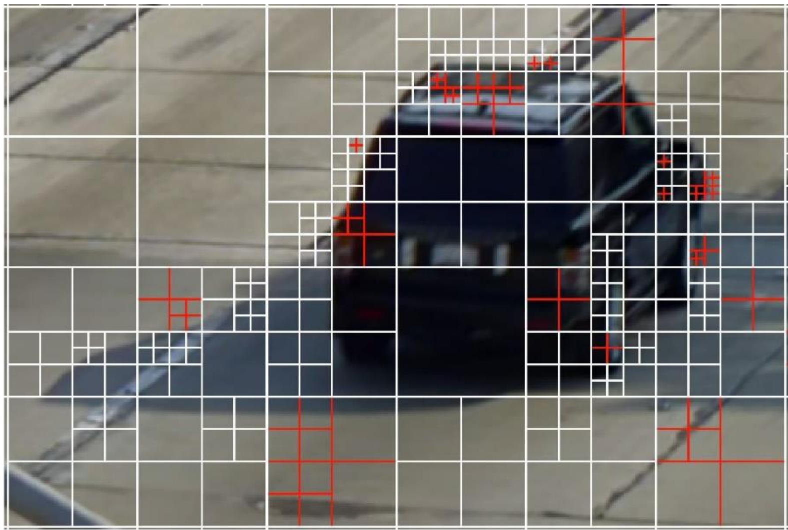
Since the encoder and decoder independently generated the same candidate list (using the same information from neighboring blocks), the encoder simply signaled to the decoder which candidate was chosen from the list, for example sending an index value of 2 to signal the second candidate from the list (e.g., the candidate from A_1). This reduced data because the encoder did not need to transmit the candidate list or even an entire 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 index. ASUS-1004, ¶37; ASUS-1003, ¶¶CA10-11.

A PMV candidate list with redundant candidates increased the list index size, requiring more bits. Thus, removing duplicates was known in the art, and it was common to do so by comparing a subset of candidates, rather than every possible pair of candidates, to reduce computation. For example, for H.264, it was known to "avoid duplicate[s]" by comparing new candidates to those already in the candidate list, which comprised a "subset" of candidates from blocks within an allowed distance, rather than to the full set of previously-coded blocks. ASUS-1004, ¶¶70-71. For H.265, a July 2011 contribution by Nakamura (explained below) proposed "[r]emov[ing] candidates with the same motion information" (ASUS-1007, Fig. 1) and comparing a subset of two candidates (i.e., "pairs") without having to compare the full set of neighboring candidates. ASUS-1007, 000003, Table 1, Fig. 1; ASUS-1003, ¶¶CA12-14.

B. H.264 to H.265

H.264 was published in 2003 and was widely adopted. In 2004, work began on improvements to H.264, and in 2010, the formal standardization process began for the successor standard H.265 ("HEVC"). H.265 used H.264 as a starting point; both shared the same architecture, with block-based motion prediction as explained above. ASUS-1003, ¶¶CA15-16.

H.265 development introduced terminology for a type of block called a coding unit ("CU"), analogous to H.264 macroblocks and divisible into smaller CUs and prediction units ("PU") to fit visual patterns in the frame. ASUS-1019, 000005; ASUS-1020, 000002 (showing blocks subdivided into smaller CUs and PUs):



ASUS-1003, ¶CA17.

Motion vectors operated on PUs. In its simplest case, a CU was commensurate with a PU when the entire block had uniform motion, such as large blocks of road in the above image. H.265 submissions included partition modes for dividing a CU into PUs when portions of the block had different motion. ASUS-1018, ¶3, Fig. 2 (showing various partition modes):

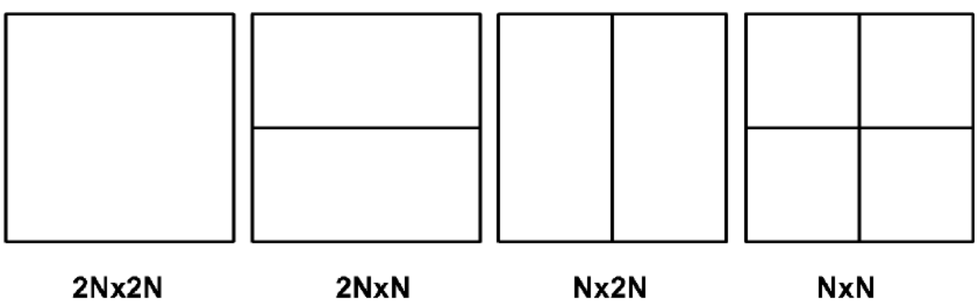


Fig. 2

ASUS-1003, ¶¶CA18-19.

II. THE ’714 PATENT

A. Overview

The ’714 patent purports to improve on H.264 and early H.265 drafts, which it admits pre-dated the patent. ASUS-1001, 1:40-42, 2:21-25. The patent reduces the number of candidates in the MVP candidate list “by performing a limited number of motion information comparisons between candidate pairs to remove redundant candidates rather than comparing every available candidate pair.” ASUS-1001, 4:18-27. In particular, the patent determines a “subset” of spatial candidates and compares a selected candidate with the subset, rather than the full set of candidates. ASUS-1003, ¶¶CA20-22.

However, the prior art already included techniques for removing redundant MVP candidates without comparing each pair from the full set of candidates. ASUS-1003, ¶CA23.

B. Prosecution History

While prosecuting the ’714 patent family in the U.S., the applicant also filed an European counterpart EP2774375. ASUS-1017, 000177; ASUS-1015, 000007. An extended European search report issued March 2016, leading to a rejection based on Nakamura. ASUS-1017, 000246-256. Patent Owner (“PO”) did not dispute that Nakamura was prior art, or that Nakamura taught the claims, and instead amended the claims with lengthy limitations making claim 1 *three pages* *long* in the EU counterpart. ASUS-1017, 000335-351; ASUS-1003, ¶¶24-26.

Subsequently, PO submitted an IDS to the USPTO citing the European search report and Nakamura during prosecution of the ’833 patent (parent of the ’714 patent). ASUS-1015, 00308. The Examiner did not consider this IDS before the Notice of Allowance. ASUS-1015, 000476-477 (mailed 12/20/2016); ASUS-1015, 000483. The ’833 patent was allowed without any substantive discussion of Nakamura or amendments like those made in the EP counterpart. ASUS-1015, 000378; ASUS-1003, ¶CA27.

Likewise, the Examiner allowed the ’714 patent without substantive discussion of Nakamura, citing as reasons for allowance “limitations analogous to the claims of” the ’833 patent for comparing motion information “without making a comparison of each pair from the set of spatial motion vector prediction candidates[.]” ASUS-1002, 000118 (Notice of Allowance). However, the European Patent Office had found a similar limitation to be taught by Nakamura (ASUS-1017, 000253), which PO did not dispute. ASUS-1003, ¶¶CA28-29.

C. Priority Date

This Petition demonstrates invalidity as of November 4, 2011, the earliest priority cited on the face of the ’714 patent[2]. ASUS-1003, ¶CA30.

III. IDENTIFICATION OF CHALLENGE

A. Statutory Grounds

Petitioner requests *inter partes* review and cancelation of the Challenged Claims on the following grounds[3]:

Grounds	Claims	Statutory Basis	Prior Art
1	1-3, 5-10, 12-17, 19-24, 26-30	§103	Rusert and Karczewicz
2	1-30	§103	Rusert, Karczewicz, and Lin

ASUS-1003, ¶¶CA32-34, CB32-34.

Grounds 1 and 2 are further supported by the expert testimony of Joseph

[2] Petitioner reserves the right to challenge the ’714 patent’s priority in other proceedings.

[3] The Board previously instituted IPR on the same Challenged Claims of the ’714 patent in IPR2024-00604 and IPR2024-00605 based on the same prior art and obviousness contentions as those presented in Grounds 1 and 2 of this Petition. ASUS-1052-1053.

Havlicek.[4] *See* ASUS-1003.

The Grounds render the challenged claims obvious because any differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a person of ordinary skill in the art ("POSITA") to which the subject matter pertains. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007).

B. Relied-Upon Prior Art[5]

[4] Dr. Havlicek's declaration incorporates as Appendices A and B the declarations of Dr. Charles D. Creusere. *See* ASUS-1003, ¶4, Appx-A ("CA"), Appx-B ("CB"). Dr. Havlicek carefully reviewed Dr. Creusere's declarations and explains that he has adopted Dr. Creusere's analysis and opinions as his own. *Id.* Rather than repeat Dr. Creusere's earlier declaration testimony, Dr. Havlicek focuses his testimony by referring to Dr. Creusere's declarations in addressing the background of the '714 patent's technology and the obviousness of the Challenged Claims in Grounds 1 and 2. *Id.* Citations with paragraph numbers labeled CA refer to paragraphs in Dr. Creusere's declaration submitted in Appendix A and citations with paragraph numbers labeled CB refer to paragraphs in Dr. Creusere's declaration submitted in Appendix B.

[5] Pre-AIA 35 U.S.C. §102 applies. MPEP §2159.02.

1. Rusert (ASUS-1004)

U.S. Patent Application Publication 2011/0194609 was filed February 7, 2011, published August 2011, and is prior art under §§102(a) and (e). ASUS-1003, ¶¶CA35-36.

Rusert teaches a method for selecting a list of PMV candidates called PMV_CANDS[6] and removes duplicates by comparing new candidates to a "*subset* of the set of previously coded motion vectors[.]" *E.g.*, ASUS-1004, ¶¶11-13; ASUS-1003, ¶¶CA37-38.

Rusert is analogous art in the same field (video encoding and decoding, *infra* §IV). Rusert teaches "video encoding" and "decoding." ASUS-1004, ¶1, ¶¶23-25; ASUS-1003, ¶CA39.

2. Karczewicz (ASUS-1005)

U.S. Patent Application Publication No. 2011/0249721 was filed April 11, 2011, published October 2011, and is prior art under §§102(a) and (e). ASUS-

[6] Rusert occasionally misspells this as "PMV_SANDS." This is a clear typo. "PMV_SANDS" and "PMV_CANDS" refer to the same list. *E.g.*, ASUS-1004, ¶39, ¶¶42-44. Where Rusert refers to "PMV_SANDS," the provisional correctly states "PMV_CANDS." ASUS-1004, ¶6; ASUS-1012, 000009-10; ASUS-1003, ¶CA38.

1003, ¶¶CA40-41.

Karczewicz teaches aspects of H.265 drafts including terminology for coding and PUs. ASUS-1005, ¶¶32-35. Karczewicz is analogous art in the same field (*infra* §IV). Karczewicz teaches techniques for "coding video data." ASUS-1005, Abstract; ASUS-1003, ¶¶CA42-43.

3. Lin (ASUS-1006)

U.S. Patent Application Publication No. 2014/0092981 claims priority to provisional 61/500,903, filed June 24, 2011, and is prior art under §102(e). Lin is entitled to this priority date because the relied-upon subject matter is described in Lin's provisional; also, claim 1 is supported by the provisional's written description, which enables a POSITA to make/use Lin's claimed invention. *Dynamic Drinkware, LLC v. Nat'l Graphics, Inc.*, 800 F.3d 1375, 1381 (Fed. Cir. 2015).

Lin's Claim 1	**Lin's Provisional**
1. A method of deriving a motion vector predictor (MVP) for a current block in an Inter, Merge, or Skip mode, the method comprising:	ASUS-1013, 000007, 000009,000012
determining neighboring blocks of the current block, wherein an MVP candidate set is derived from MVP candidates associated with the neighboring blocks;	ASUS-1013, 000016 (Fig. 6)

determining at least one redundant MVP candidate according to a non-MV-value based criterion;	ASUS-1013, 000009-11
removing said at least one redundant MVP candidate from the MVP candidate set; and	ASUS-1013, 000009-11
providing a modified MVP candidate set, wherein the modified MVP candidate set corresponds to the MVP candidate set with said at least one redundant MVP candidate removed.	ASUS-1013, 000008-11

ASUS-1003, ¶¶CA44-45. Ground 2 below shows citations to Lin and Lin's provisional.

Lin teaches removal of redundant MVPs for divided CUs, which is relevant to claim 4. ASUS-1006, ¶25. Lin is analogous art in the same field (*infra* §IV). Lin teaches techniques "for motion vector coding." ASUS-1006, ¶2; ASUS-1003, ¶¶CA47-48.

C. Standing

ASUS certifies that the '714 Patent is available for IPR. The present petition is being filed within one year of service of a complaint against ASUS. ASUS is not barred or estopped from requesting this review challenging the Challenged Claims on the above-identified grounds.

IV. FIELD AND LEVEL OF SKILL

The field of the '714 patent is video encoding/decoding. ASUS-1001, 1:40-

42, 2:21-25. A POSITA at the time of the alleged invention of the ’714 patent would have had (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. ASUS-1003, ¶¶CA57-59.

V. CLAIM CONSTRUCTION

Claims should be construed according to the ordinary and customary meaning as understood by a POSITA. 37 C.F.R. §42.100(b).

A. “spatial motion vector prediction candidate”

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. ASUS-1003, ¶CA62.

The ’714 patent states “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.” ASUS-1001, 3:9-14. A “*candidate* motion vector” uses “one or more neighbour blocks and/or other blocks of the current block *in the same frame*” and “represents the motion vector of one or more already encoded block.” ASUS-1001, 12:51-59; ASUS-1003, ¶CA63.

B. “temporal motion vector prediction candidate”

A POSITA would have understood a “temporal motion vector prediction candidate” to mean a candidate motion vector obtained from a previously-encoded frame. ASUS-1003, ¶CA64.

The ’714 patent states that a “temporal motion vector *prediction* is a prediction obtained on the basis of one or more blocks of a frame different from the current frame.” ASUS-1001, 3:12-14. Furthermore, the patent states 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.” ASUS-1001, 12:63-13:3; ASUS-1003, ¶CA65.

C. “the block”

Limitation [1b] recites “the block,” which could be interpreted to refer to either (a) the “block of pixels” introduced in [1a], or (b) the block from which the first spatial motion vector candidate is obtained. For purposes of this IPR, Petitioner provides disclosure under both constructions. In IPR2024-00604 and IPR-2024-00605, the PTAB construed “the block” in limitation [1b] to mean “the block associated with the first special motion vector prediction candidate.”

D. “a subset of … candidates”

Limitation [1b] recites “a subset of … candidates,” which means a subset of one or more candidates. The claims confirm that the subset may comprise one candidate. Limitation [1c] compares motion information of a potential candidate with motion information of "candidates in the determined subset" of limitation [1b]; dependent claim 3 further specifies that the potential candidate is compared with "at most one other" candidate. The specification includes embodiments where the subset is a single candidate. ASUS-1001, 15:50-16:39 (e.g., block A1 is compared with only block B1), Fig. 8b; ASUS-1003, ¶CA67.

VI. CHALLENGED CLAIMS

A. Grounds 1 and 2

1. Motivation to Combine and Reasonable Expectation of Success

Ground 1. Rusert teaches all limitations of the independent claims. Rusert uses H.264 terminology and teaches the blocks that serve as PUs in H.264. To the extent the claims require H.265 PUs, those were well-known and, for example, taught by Karczewicz. A POSITA would have been motivated to apply Rusert's teachings to H.265 PUs and related concepts, as explained below. ASUS-1003, ¶CA71.

Rusert does not use H.265 terminology. Nonetheless, Rusert explains that, while its examples 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." ASUS-1004, ¶116. Karczewicz teaches the concept of PUs in H.265, which was emerging as the successor standard to H.264—both were drafted by ITU-T—sharing the same concepts of block-based video encoding with predicted motion vectors. ASUS-1005, ¶32, ¶¶35-37, ¶66, ¶71; ASUS-1006, ¶5. Karczewicz teaches H.265 concepts including PUs. Therefore, the art provides express motivation to apply Rusert's teachings to other standards including H.265, as taught by Karczewicz. ASUS-1003, ¶¶CA71-74.

This would have combined prior art elements according to known methods to yield predictable results, e.g., combining Rusert's teachings for generating/de-duplicating MVP candidate lists with H.265 concepts, including PUs and related information for motion vectors. In Rusert's H.264 examples, motion vectors and motion prediction operate on blocks. ASUS-1004, ¶¶2-5, ¶36, ¶43. As Karczewicz explains, for H.265 drafts, motion vectors and motion prediction operated on blocks called PUs. ASUS-1005, ¶¶33-36, ¶¶64-66. Karczewicz also teaches types of information conveyed by motion vectors. ASUS-1005, ¶35. A POSITA would have been motivated to apply Rusert's known techniques for selecting PMV candidates to PUs, to improve the similar draft H.265 standard in the same way that Rusert explained for H.264. This application would have been a simple substitution of Karczewicz's PU teachings for Rusert's block teachings, which would have yielded several advantages, including improved coding efficiency as taught by both references. ASUS-1004, ¶7; ASUS-1005, ¶6; ASUS-1003, ¶¶CA75-

79.

Additionally, a POSITA would have found motivation in the similarity of the references. Both are directed to ITU video-coding standards. ASUS-1004, ¶116; ASUS-1005, ¶¶32-33. Karczewicz explains that "[i]n general, a CU" of H.265 "has a similar purpose to a macroblock of H.264" and in its simplest case, a CU was a PU. ASUS-1005, ¶33, ¶35; ASUS-1003, ¶CA78.

A POSITA would have been motivated to combine teachings from Rusert's embodiments and background section regarding H.264 with each other. Rusert explains that its embodiments were provided in the context of H.264. ASUS-1004, ¶¶3-4,¶67, ¶77, ¶116; ASUS-1003, ¶CA80.

Ground 1 relies on Rusert's teachings regarding PMV_CANDS, for which Rusert teaches two options: initializing and updating PMV_CANDS dynamically for each block or using a sliding window. ASUS-1004, ¶41. A POSITA would have been motivated to use the first option (for each block) because Rusert presents it as the first of a finite number of options (two). ASUS-1003, ¶CA81.

Karczewicz discusses H.264 and H.265 and applies its teachings to both. Therefore, Karczewicz's teachings regarding additional H.265 features would not have dissuaded a POSITA from combining Karczewicz with Rusert. ASUS-1005, ¶5, ¶38. H.265 does not prevent H.264 teachings from being applied, or vice-versa. ASUS-1003, ¶CA82.

The combination would have had predictable results. Rusert already applies its teachings to block-based encoding/decoding. ASUS-1004, ¶2, ¶11. A PU is a type of block. ASUS-1005, ¶35. Rusert's concepts were readily applicable to PUs, and the combination would have had the predictable result of selecting PMV candidates (as Rusert teaches) for PUs (as Karczewicz teaches). ASUS-1004, ¶116; ASUS-1003, ¶CA83. Therefore, the combination would not have changed the principle of operation for Rusert or Karczewicz because it applies their teachings in the manner taught by each. Applying H.265 teachings to Rusert, and vice versa, was consistent with Rusert's statement that its principles are applicable to other standards. ASUS-1004, ¶116; ASUS-1003, ¶CA84.

A POSITA would have had a reasonable expectation of success combining Rusert and Karczewicz. The combination applies teachings according to their known purposes, in a conventional manner. The teachings are complementary aspects of block-based video encoding from H.264 and H.265. ASUS-1004, ¶116. Karczewicz complements Rusert by teaching H.265 terminology and concepts. ASUS-1005, ¶32. Ground 1 does not modify Rusert or Karczewicz in a way that would render either inoperative. To the contrary, the similarities of the architectures would have given a POSITA a reasonable expectation of success in combining their teachings. ASUS-1004, ¶11; ASUS-1005, ¶2; ASUS-1003, ¶¶CA85-86.

A POSITA would have been more than capable of applying Karczewicz's teachings to Rusert and vice versa because it would simply apply Rusert's teachings from H.264 to H.265. Motion estimation had been commonplace since the 1990s; H.264 introduced MVP by the early-mid 2000s. These were basic aspects of video-coding a POSITA would have been knowledgeable about. *Supra* §IV; ASUS-1003, ¶CA87.

The motivation to combine and reasonable expectation of success for Ground 2 is explained for Claim 4.

2. Claim 1

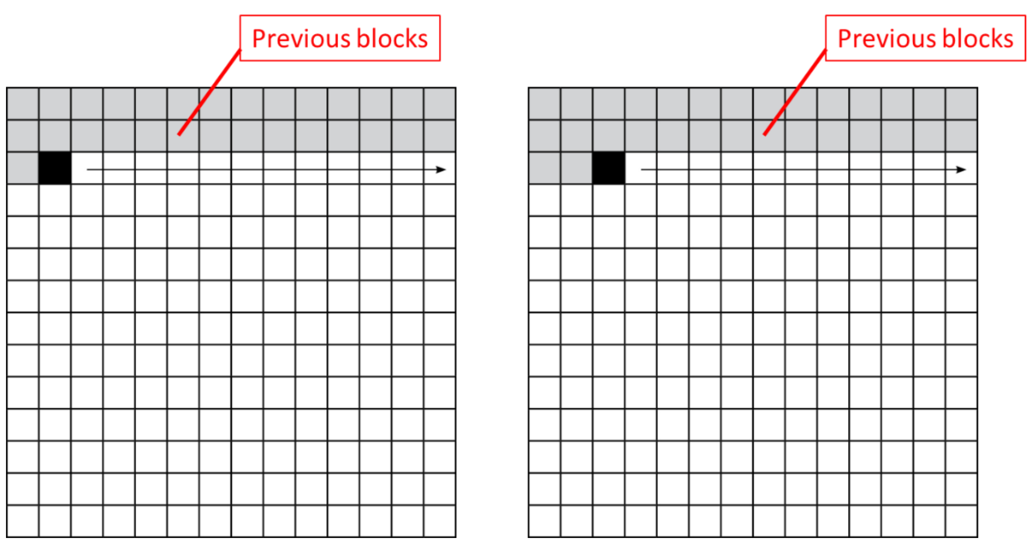
[1pre]	**A method**

Ground 1 teaches [1pre], e.g., "a *method* of selecting PMV candidates" comprising elements explained below. ASUS-1004, Abstract, ¶1, ¶11; *infra* §§VI.A.2[1a]-[1e]; ASUS-1003, ¶¶CA99-100.

[1a]	**selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for a block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates and is utilized to identify motion vector prediction candidates of which one spatial motion vector prediction candidate from the motion vector prediction list is signaled as the motion information for the prediction unit;**

Ground 1 teaches [1a]. ASUS-1003, ¶¶CA101-118.

Selecting a first spatial motion vector prediction candidate (PMV candidate) **from a set of spatial motion vector prediction candidates** (set of previously-coded motion vectors) **for a block of pixels**. Rusert teaches "selecting… PMV candidates" for a current block from a "*set* of previously coded motion vectors that were used for previous blocks." ASUS-1004, ¶¶11-12, ¶15, ¶¶24-25, ¶39, ¶44, ¶¶51-66, ¶113, Fig. 6. As Rusert iterates through blocks of pixels in a frame, each block will have its own unique set of previously-coded motion vectors from which to select a PMV candidate because after a block is encoded/decoded, the set of previously-coded motion vectors increases, as illustrated below. ASUS-1004, ¶2 ("pixel blocks"), ¶¶11-12, ¶36, ¶59, Fig. 3g; ASUS-1003, ¶¶CA102-104:



Rusert's PMV candidates (ASUS-1004, ¶3) comprise spatial motion vector prediction candidates obtained from previously-encoded blocks in the current frame. *Supra* §V.A. They include "spatially neighboring motion vectors" (ASUS-1004, ¶6, ¶¶3-5) and are included in PMV_CANDS, which "comprise[s] *spatial* … *neighbors* of the current block" in the current frame. ASUS-1004, ¶67, ¶¶4-6; ASUS-1003, ¶CA103.

Selecting a first… candidate… as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list (PMV_CANDS). Rusert teaches search patterns for "an outwards going scan… around the current block" for selecting PMV candidates to potentially be included in PMV_CANDS. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n (showing numerical scan order around current block "." from block 1 onwards, with directional annotation added for 3n):

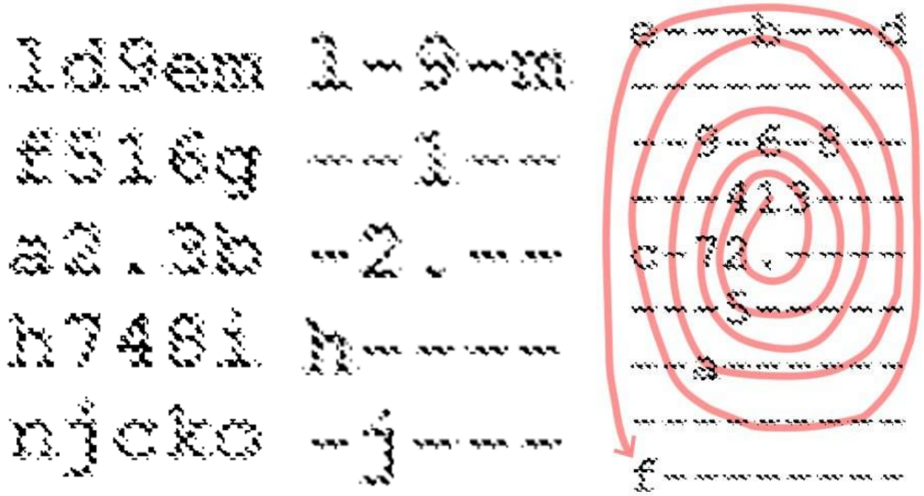


Fig. 3b Fig. 3c Fig. 3n

ASUS-1003, ¶¶CA105-107.

Following these search pattern sequences, Rusert visits a previously-coded block and selects the motion vector for that block as a candidate for potential inclusion in PMV_CANDS. ASUS-1004, ¶44, ¶¶51-66; *infra* §VI.A.2[1c]-[1d] (explaining evaluation for potential inclusion in PMV_CANDS); ASUS-1003, ¶CA107.

PMV_CANDS is **a motion vector prediction list for a prediction unit of the block of pixels**. It is a list of predicted motion vectors "used for coding a motion vector associated with a current… block." ASUS-1004, ¶41, ¶4, ¶¶39-42. PMV_CANDS is "dynamically generated specifically for the current… block[.]" ASUS-1004, ¶41. "[B]efore a block is processed, a PMV_CANDS list is initialized and then updated with… motion vectors." *Id*. When PMV_CANDS is updated, i.e., when "one or more motion vectors are added" (ASUS-1004, ¶39), it comprises a subset of the set of previously-coded motion vectors that have been included in PMV_CANDS to that point. ASUS-1004, ¶¶4-5, ¶¶36-39, ¶¶43-44, ¶¶51-66. When PMV_CANDS is complete, a PMV that "is used to predict a [motion vector]… is signaled" using an index "to select a particular PMV candidate… from PMV_CANDS[.]" ASUS-1004, ¶36; ASUS-1003, ¶CA108.

The prediction unit of Rusert's block of pixels is the block itself because that is the unit for which a motion vector is assigned for motion prediction. ASUS-1004, ¶¶2-4, ¶43. Rusert provides a motion vector for each "8x8 pixel block," also called a "sub-block" because it is a portion of a "macroblock." ASUS-1004, ¶36. Rusert scans neighboring blocks because motion vectors are assigned based on blocks, which are the PUs in Rusert's teachings. *See* ASUS-1004, ¶¶50-67, Figs. 3a-3n; ASUS-1003, ¶CA109.

Additionally, Ground 1 combines Rusert's "block" teachings (following H.264 terminology) to PUs (in H.265 terminology). *Supra* §VI.A.1; ASUS-1004, ¶116. As Karczewicz explains, H.265 introduced terminology for a type of block called "prediction unit[s.]" ASUS-1005, ¶¶33-36; *supra* §I. "In general, a CU has a similar purpose to a macroblock of H.264" (ASUS-1005, ¶33), which in its simplest case is commensurate with a PU but may also be divided into multiple

PUs. ASUS-1005, ¶35. "[T]he PU may include data defining a motion vector for the PU" that is used for "prediction using a PU[.]" ASUS-1005, ¶¶35-36, ¶66; ASUS-1003, ¶¶CA110-111. In short, Karczewicz explains that H.265 assigned motion vectors based on a type of block called a PU. ASUS-1005, ¶¶33-36, ¶64. Rusert explained its teachings based on the blocks for which motion vectors were assigned (ASUS-1004, ¶¶2-5, ¶36, ¶43). Therefore, it was obvious to apply Rusert's teachings to PUs, with PMV_CANDS being a motion vector prediction list for a PU of the block of pixels. ASUS-1005, ¶¶33-36, ¶66; ASUS-1003, ¶¶CA110-112.

Where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates. PMV_CANDS is a "list of PMV candidates." ASUS-1004, ¶37. Each candidate is a motion vector (ASUS-1004, ¶11, ¶¶24-25, ¶¶39-41) including "x and y components" (ASUS-1004, ¶106, ¶36, ¶¶91-94, ¶100). Therefore, Rusert's PMV_CANDS list includes motion information, including motion vectors and their x and y components, of the PMV candidates. ASUS-1004, ¶¶2-3; ASUS-1003, ¶CA113.

Additionally, Karczewicz teaches "the 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[.]" ASUS-1005, ¶35. It was obvious to include this PU information in PMV_CANDS because the combination relies on Karczewicz's PU teachings. *Supra* §VI.A.1; ASUS-1003, ¶CA114.

PMV_CANDS **is utilized to identify motion vector prediction candidates of which one spatial motion vector prediction candidate from the motion vector prediction list is signaled as the motion information for the prediction unit**.[7] Rusert uses an index to signal "a particular PMV candidate… from… PMV_CANDS" as the "predicted motion vector" for a block. ASUS-1004, ¶¶35-37, ¶4 ("*signal a PMV to* be used out of… PMV_CANDS"), ¶75, ¶88, Table 1. This index is transmitted to a decoder that uses this "transmitted index" to "determine the PMV… as used in the encoder" from PMV_CANDS to "reconstruct [a motion vector]." ASUS-1004, ¶37, ¶¶88-102 (exemplary index codes). The index is transmitted as the motion information for the block. ASUS-1004, ¶¶35-37; *infra* §VI.A.2[1e]. Ground 1 applies Rusert's block-based teachings to PUs. *Supra* §VI.A.1; ASUS-1003, ¶¶CA115-116.

Rusert Figs. 2a-2b show PMV candidate 220 being signaled as the motion information for a block, using index 250. ASUS-1004, ¶36. A decoder uses index 250 to identify PMV 220 as the PMV candidate to be used for the current block.

---

[7] The '714 patent admits this was known. ASUS-1001, 3:60-66; ASUS-1003, ¶CA115n.2.

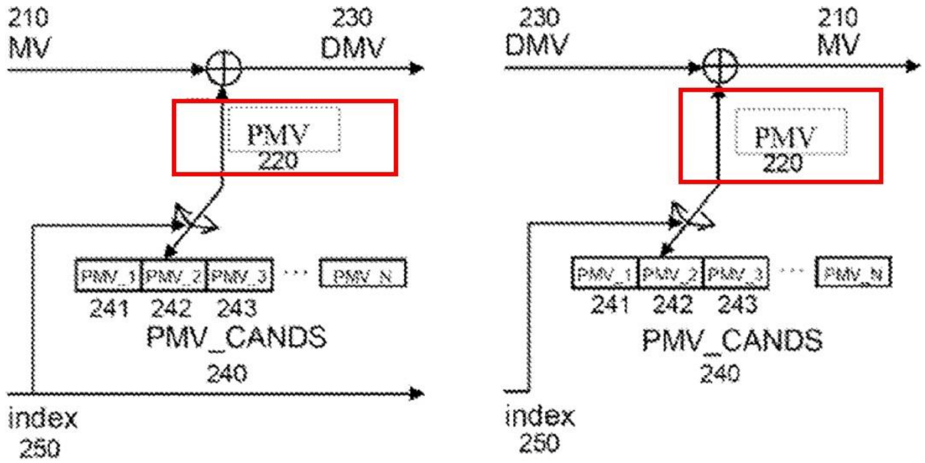


Fig. 2a Fig. 2b

ASUS-1003, ¶¶CA117-118.

[1b]	**determining a subset of spatial motion vector prediction candidates based on a location of the block associated with the first spatial motion vector prediction candidate;**

Ground 1 teaches [1b]. ASUS-1003, ¶¶CA119-131.

Determining a subset of spatial motion vector prediction candidates (e.g., based on Rusert's scan of previous PMV candidates within an allowed distance and pre-defined number). Rusert "select[s] a set of PMV candidates as a *subset* of the set of previously coded motion vectors that were used for previous blocks[.]" ASUS-1004, ¶¶11-12, ¶¶24-25; *supra* §V.D. "[T]he 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[.]" ASUS-1004, ¶15, ¶37, Figs. 2a-2b; ASUS-1003, ¶¶CA120-121.

Rusert implements the subset of PMV candidates using an outward scan of blocks having previously-coded motion vectors, starting from the current block and staying within an allowed distance. ASUS-1004, ¶¶4-5, ¶¶36-39, ¶¶43-44, ¶¶51-66. As the scan progresses outwards, the subset includes the previously-coded PMV candidates obtained from blocks in previous locations of Rusert's scan. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n. For example, when the selected first spatial motion vector prediction candidate is obtained from the third block in Rusert's scan sequence, the subset of spatial motion vector prediction candidates comprises the candidates from previous scan locations (the first and second blocks). *Id.*

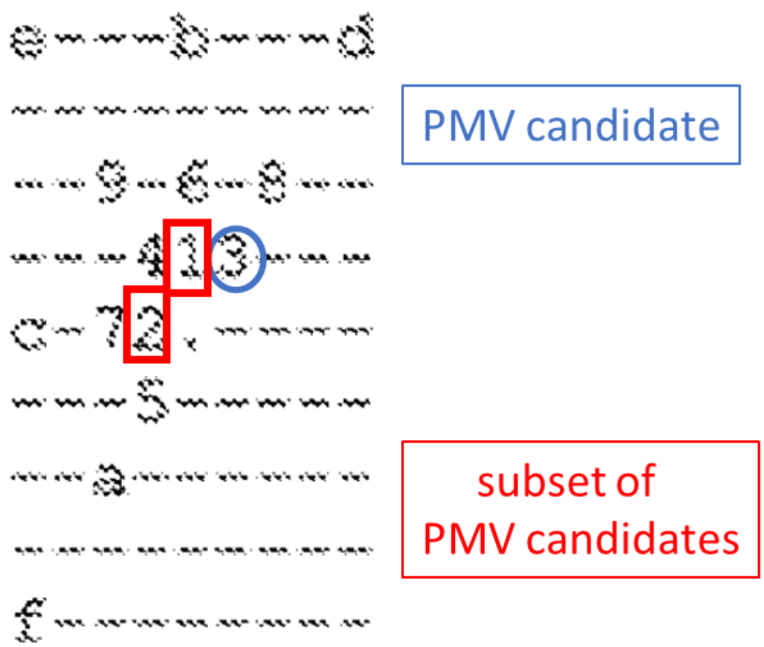


Fig. 3n

ASUS-1003, ¶¶CA121-122.

Additionally, Rusert terminates the scan "as soon as a pre-defined number of unique PMV candidates have been found." ASUS-1004, ¶48. Even PMV candidates that are within "a certain distance" and part of "a predetermined scan pattern" are not considered if the "pre-defined number of unique PMV candidates have been found." ASUS-1004, ¶¶44-49. Rusert improves coding efficiency by using subsets. ASUS-1004, ¶¶12-13. Therefore, Rusert's scan of previously-coded motion vectors is further cabined into a subset because it is limited to this predefined number of PMV candidates. For example, while Fig. 3n includes a scan of up to 15 blocks, the scan terminates when a pre-determined number of candidates are obtained (e.g., 7), without using the remaining candidates in the scan. ASUS-1004, ¶48, ¶107; ASUS-1003, ¶¶CA123-124.

As the scan progresses outwards and the subset of PMV candidates is updated, the subset is stored as a list of PMV candidates called PMV_CANDS. ASUS-1004, ¶¶37-41, ¶¶44-49, ¶¶51-66; *infra* §VI.A.2.[1d]. The subset is determined for reducing the number of candidates, as explained for [1c]. ASUS-1003, ¶CA123.

Based on a location of the block associated with the first spatial motion vector prediction candidate. Under the construction that "the" block is the block from which the selected first candidate is obtained, Rusert teaches this limitation. *Supra* §V.C. A "position" of a block of a PMV candidate is represented as (xpos, ypos), i.e., a location of the block associated with the first spatial motion vector prediction candidate. ASUS-1004, ¶¶51-52. Based on the block's location, the subset of PMV candidates with which the PMV candidate is compared includes the candidates of blocks located in the scan pattern up to the PMV candidate. ASUS-1004, ¶44. For example, in Fig. 3n, the PMV candidate for block "3" is compared with a subset of PMV candidates for blocks "1" and "2." ASUS-1004, ¶44, ¶¶65-66, Fig. 3n:

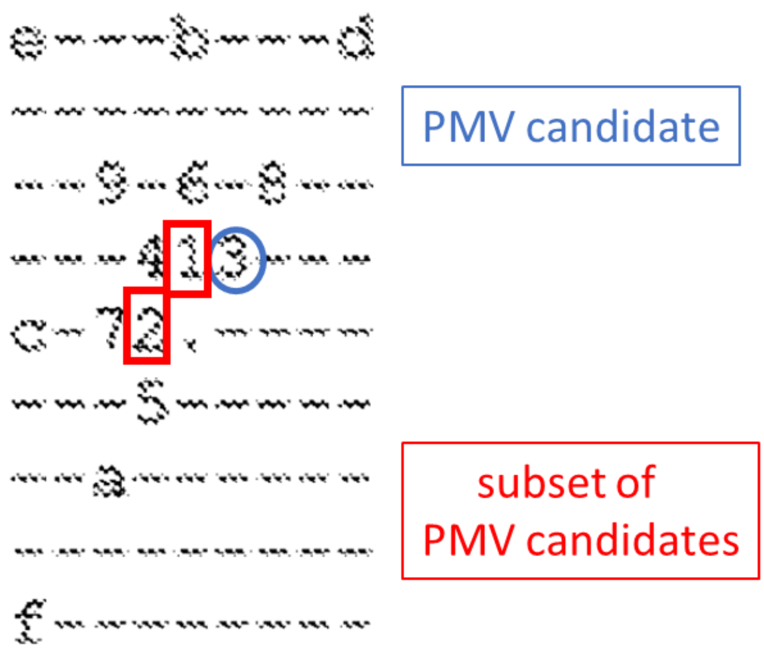


Fig. 3n

ASUS-1003, ¶CA130.

Rusert teaches this limitation even if, "the block" refers to the "block of pixels" in [1a] and therefore refers to the current block for which candidates are being analyzed. *Supra* §V.C. Because the subset of PMV candidates is based on "an outwards going scan… around the current block" (ASUS-1004, ¶¶43-44), the subset of blocks that are scanned and the corresponding subset of spatial motion vector prediction candidates from those blocks are based on the location of the current block. ASUS-1004, ¶44; ASUS-1003, ¶¶CA125-126.

Furthermore, the subset of PMV candidates is based on "an allowed distance from the current block and an allowed position." ASUS-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. ASUS-1004, ¶¶11-17, ¶¶24-25, ¶¶43-44, ¶¶51-66, Figs. 3a-3n. Rusert teaches exemplary distance values[8] based on the location of the current block ("."). ASUS-1004, ¶47, ¶13, ¶43, Figs. 3a, 3d, 3f, 3g:

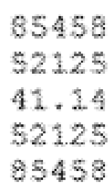


Fig. 3a

Fig. 3d

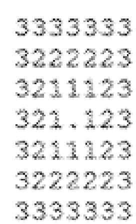


Fig. 3f

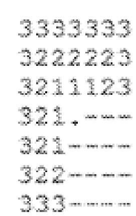


Fig. 3g

Only candidates from blocks with allowed positions relative to the current block are included. ASUS-1004, ¶¶15-16, ¶59 (excluding "blocks to the right

---

[8] "Euclidean distance" is occasionally misspelled "Euclidian distance." ASUS-1004, ¶44, ¶78, ¶87; ASUS-1003, ¶CA127n.4.

and below the current block”), ¶65; ASUS-1003, ¶¶CA127-129.

[1c]	**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 pair from the set of spatial motion vector prediction candidates;**

Ground 1 teaches [1c]. ASUS-1003, ¶¶132-143.

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. Rusert teaches three comparisons for excluding “unnecessary” candidates; each satisfies [1c]. ASUS-1004, ¶21, ¶¶71-72; *supra* §VI.A.2[1b]; ASUS-1003, ¶CA133.

First, Rusert compares whether “the PMV candidate is a duplicate of *another* PMV candidate” in the subset of PMV candidates, which is stored as a list in PMV_CANDS. ASUS-1004, ¶21, ¶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… skip the new vector” thereby “reducing the number of candidates[.]” ASUS-1004, ¶71, ¶62. A potential PMV candidate is selected and compared with each PMV candidate in the subset of preceding PMV candidates in Rusert's scan sequence, which are stored in PMV_CANDS. *Supra* §VI.A.2[1b]. If the x and y components of two motion vectors are the same, they are duplicates and the potential candidate is excluded. *Id.*; ASUS-1003, ¶CA134.

Second, Rusert compares whether "the PMV candidate is… within a threshold distance of an existing PMV candidate" in the subset of candidates, e.g., using "a similarity measure" such as Euclidean distance to calculate whether the difference between the x and y components of their motion vectors is "smaller than a pre-defined threshold"; if so, the potential candidate is removed/skipped. ASUS-1004, ¶21, ¶72, ¶87. The distance between duplicate PMV candidates is zero. ASUS-1003, ¶¶CA135-136, ¶135n.5.

Third, Rusert compares whether "at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits" (ASUS-1004, ¶21), thereby "removing PMV candidates" that "will never be used" because another PMV candidate in the subset facilitates "a bit sequence that is shorter or of the same length compared for all possible motion vectors." ASUS-1004, ¶¶90-94. When the x and y values of an existing candidate's motion vector can be added to the x and y values of a difference motion vector ("DMV")[9] to yield the x and y values of the selected candidate while using fewer overall bits, Rusert excludes the selected

[9] *Supra* §I.

candidate. ASUS-1004, ¶95 (teaching exemplary PMV/DMV combinations requiring fewer bits); ASUS-1003, ¶CA136.

All three teachings compare a potential candidate with a subset of preceding candidates in Rusert's scan. Since preceding candidates have smaller index values, they would be signaled more efficiently than later duplicates, and Rusert improves efficiency by determining this subset and comparing potential new candidates to the subset. ASUS-1004, ¶¶88-98; ASUS-1003, ¶CA137.

All three teachings compare motion information of the PMV candidates, including the x and y components of their motion vectors. This is motion information of the PMV candidates because they describe the "motion of pixel blocks across frames[.]" ASUS-1004, ¶2. Motion vector coding includes an MVP combined with a DMV. ASUS-1004, ¶3, ¶¶36-37; *supra* §VI.A.2.1[a] (explaining motion information); ASUS-1003, ¶CA137.

Rusert teaches or at least suggests performing three comparisons when evaluating whether to update PMV_CANDS with a PMV candidate. ASUS-1004, ¶71, ¶75, ¶¶84-87, ¶90; ASUS-1003, ¶CA138.

Comparing … without making a comparison of each pair from the set of spatial motion vector prediction candidates. Rusert's three comparisons (explained above) each compare motion information of a selected candidate with the subset of PMV candidates stored in PMV_CANDS, without making a comparison of each pair from the set of motion vectors used for all previous blocks. ASUS-1004, ¶¶11-13, ¶¶15-16, ¶113. Rusert teaches that the subset of PMV candidates is indeed a "*subset* of the motion vectors previously used for previous blocks." ASUS-1004, ¶¶11-12. Even within the scan order, Rusert compares potential candidates with preceding candidates already in PMV_CANDS, which are a subset of the candidates in the scan order; therefore, Rusert avoids comparing each pair of candidates from the scan order. *See id.*; ASUS-1003, ¶CA139.

Moreover, even within Rusert's scan patterns, Rusert teaches that a "scan may be terminated… as soon as a pre-defined number of unique PMV candidates have been found[.]" ASUS-1004, ¶48. For example, Rusert teaches "us[ing] a maximum of four candidates in the [PMV_CANDS] list" or "us[ing] seven" as the maximum. ASUS-1004, ¶107. Therefore, Rusert does not compare each possible pair of candidates from the set of all previously-coded blocks, or even each possible pair of blocks within the scan order. Rusert thus teaches Claim 1 with the recited "set" being either (a) the previously-coded motion vectors for that frame, 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. ASUS-1003, ¶CA142.

Additionally, because the subset is limited to candidates having "an allowed distance" and "position[,]" candidates from previous blocks outside the allowed distance/position are not in PMV_CANDS and, therefore, are not compared. ASUS-1004, ¶15; *supra* §VI.A.2[1b]. Furthermore, Rusert teaches scan patterns that exclude certain blocks within the allowed distance. ASUS-1004, ¶¶64-65, Figs. 3m, 3n. Given these excluded blocks, there is no comparison of each pair of motion vectors from the set of all previously-coded blocks. ASUS-1003, ¶¶CA140-141.

[1d]	**determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and**

Ground 1 teaches [1d]. ASUS-1003, ¶¶CA144-149.

As explained above, Rusert determines whether to include or exclude the selected first PMV candidate in PMV_CANDS, which stores the subset of PMV candidates, based on each of the three comparisons explained for [1c]. Rusert removes "unnecessary PMV candidates" to "ensure[] the length of the [PMV_CANDS] list is not unnecessarily long[.]" ASUS-1004, ¶¶20-21. Rusert explains that "[a] PMV candidate *may be determined to be unnecessary*" and be therefore excluded if at least one of the three comparisons is fulfilled. ASUS-1004, ¶21; *supra* §VI.A.2[1c]. A shorter PMV_CANDS list "allows the remaining PMV candidates to be signaled using shorter codes and so fewer bits[.]" ASUS-1004, ¶22, ¶90; ASUS-1003, ¶¶CA145-146.

Rusert excludes unnecessary candidates when deciding whether to add a new candidate to the subset of PMV candidates stored in PMV_CANDS. ASUS-1004, ¶21, ¶¶71-72; s*upra* §VI.A.2[1c]. As part of the "update" process whereby new candidates are added, Rusert determines whether a selected PMV candidate should be included or skipped, meaning it is excluded based on any of the three comparisons from [1c]. ASUS-1004, ¶¶71-72. Additionally, this would have been obvious because Rusert teaches the advantages of reducing the number of candidates in PMV_CANDS using the three comparisons from [1c] (ASUS-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 subset of PMV candidates stored in 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[.]" ASUS-1004, ¶90; ASUS-1003, ¶¶CA147-148.

[1e]	**causing information identifying the one spatial motion vector prediction candidate from the motion vector prediction list to be transmitted to a decoder or to be stored.**

Ground 1 teaches [1e]. ASUS-1003, ¶¶CA150-156.

Rusert causes information (e.g., an index) identifying the one spatial motion vector prediction candidate from PMV_CANDS to be stored and transmitted to a decoder. For example, Rusert teaches an index "to select a particular PMV candidate… from… PMV_CANDS[.]" ASUS-1004, ¶35. The index is stored in the video stream and transmitted "from the encoder… to the decoder[.]" ASUS-1004, ¶35, ¶150. The index identifies and signals one PMV candidate from PMV_CANDS. ASUS-1004, ¶¶35-37, ¶75, ¶¶91-92, Figs. 2a-2b; *supra* §VI.A.2[1a]. In an encoded video stream, the "index… may be sent once together with each transmitted motion vector" and the decoder, "[u]sing the transmitted index, … can determine the PMV… as used in the encoder[.]" ASUS-1004, ¶¶35-37, Fig. 1:

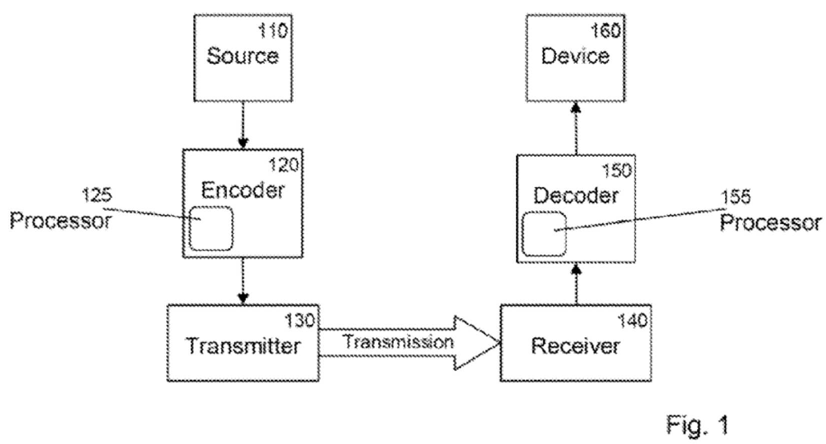


Fig. 1

ASUS-1003, ¶¶CA151-152.

Rusert teaches exemplary binary codes for the index used for transmission and storage. ASUS-1004, ¶¶88-102, Table 1:

Value	Index	Code
(−1,2)	0	1
(13, 4)	1	010
(12, 3)	2	011
(0, 2)	3	0010
(3, 4)	4	0011
(−4, 1)	5	0000
(4, 8)	6	0001

ASUS-1003, ¶CA153.

Furthermore, Rusert teaches "the respective lists of PMV candidates" are "stored in the encoder and decoder[.]" ASUS-1004, ¶35. Corresponding indices that identify candidates from those lists would have obviously been stored as well because the candidate lists serve no purpose without an index to identify a candidate, and Rusert teaches indices are used with the lists.[10] ASUS-1004, ¶35, ¶¶88-102. Ground 1 applies Rusert's teachings from "blocks" to PUs. *Supra* §VI.A.1; ASUS-1003, ¶¶CA154-155.

3. Claim 2

2.	**The method according to claim 1 further comprising selecting spatial motion vector prediction candidates from the set of spatial motion vector prediction candidates as the potential spatial motion vector prediction candidate in a predetermined order.**

[10] For example, computers convert inputs of digits to binary code, then store and transmit the binary code. ASUS-1003, ¶CA154n.6.

As explained above, Ground 1 teaches 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 (*supra* §VI.A.2[1a]) from "a plurality of blocks surrounding a current block." ASUS-1004, ¶30, ¶40, ¶¶51-66; ASUS-1003, ¶¶CA157-163.

Rusert selects candidates in a predetermined order. Rusert teaches "a simple procedure to scan the candidates in order" (ASUS-1004, ¶58) including ordered "scan patterns… for a plurality of blocks surrounding a current block[.]" ASUS-1004, ¶30, ¶40, ¶¶51-66, Figs. 3a-3n.

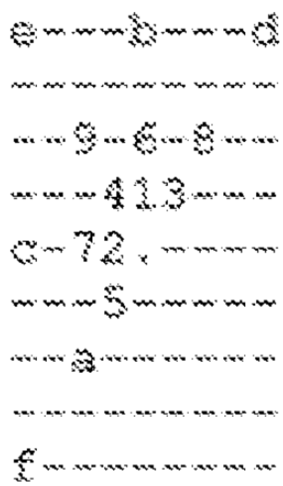


Fig. 3n

ASUS-1003, ¶CA160.

For example, Figs. 3b, 3c, 3e, 3l, 3m, and 3n scan blocks in ascending order from 1 onward, with letters representing ordering of 10 and above ("a" is 10, "b" is

11, etc.). In Fig. 3c, Rusert scans “blocks in the following order: 1, 2, 9, h, j, l, m.” ASUS-1004, ¶¶53-54, Fig. 3c:

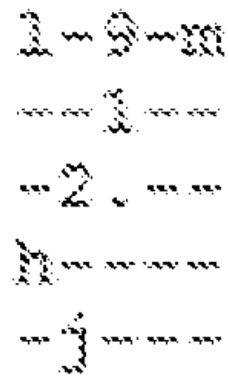


Fig. 3c

ASUS-1003, ¶CA161.

Rusert discusses known benefits/tradeoffs associated with its exemplary scan patterns and highlights Fig. 3n for its balanced compression efficiency and computation performance. ASUS-1004, ¶66. Rusert nonetheless teaches the use of each exemplary scan pattern because each example had its known applications where known benefits outweighed tradeoffs. ASUS-1004, ¶¶51-65. Thus, each scan was known and used as a predetermined pattern. ASUS-1004, ¶30; ASUS-1003, ¶CA162.

4. Claim 3

3.	**The method according to claim 1, further 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.**

Ground 1 teaches claim 3. ASUS-1003, ¶¶CA164-168. Rusert teaches three comparisons; each satisfies the comparing step explained for [1c]. *Supra* §VI.A.2[1c]; ASUS-1004, ¶21. Rusert further teaches claim 3's additional limitation of comparing motion information for **at most one other** candidate of the set. For Rusert's scan sequences (*supra* §VI.A.2[1a]-[1b]; ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n), the "PMV_CANDS list may be initialized e.g. as an empty list (zero entries)" and then "updated to include previously coded motion vectors[.]" ASUS-1004, ¶39, ¶44, ¶71; *supra* §VI.A.2[1c]. When the scan moves to update PMV_CANDS with the second PMV candidate in the sequence, that candidate is "compar[ed with] the candidates already in the list[,]" which is at most the first PMV candidate in the scan sequence. ASUS-1004, ¶21, ¶¶38-40, ¶44, ¶71. Therefore, the motion information of the second candidate in the sequence is compared with at most one other PMV candidate (the first PMV candidate). ASUS-1003, ¶¶CA165-167. In addition, Rusert's table 1 discloses a maximum code length of 2. ASUS-1004, ¶88, Table 1; ASUS-1003, ¶41. When the code length is 2, the motion information of the potential spatial motion vector prediction candidate is always compared to the motion information of at most one other spatial motion vector prediction candidate. ASUS-1004, ¶88, Table 1; ASUS-1003, ¶41.

5. Claim 4

4.	**The method according to claim 1 further comprising examining whether the 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 motion vector prediction list if the prediction unit is the second prediction unit.**

Rusert and Karczewicz teach **the method according to claim 1**. A POSITA would have been motivated to further apply Lin's teachings, which satisfy claim 4. ASUS-1003, ¶¶CA88-98, ¶¶CA169-176.

Examining whether the block of pixels is divided into a first prediction unit and a second prediction unit. Lin examines whether a block of pixels is divided. Lin teaches that, in H.265/HEVC, "[t]he basic unit for compression, termed Coding Unit (CU), is a 2Nx2N square block, and… [e]ach CU contains one or multiple Prediction Units (PUs)" with divisions "correspond[ing] to horizontal and vertical partition[s.]" ASUS-1006, ¶4; ASUS-1013, 000007, 000017. Horizontally-divided CUs have PUs of size 2NxN; vertically-divided CUs have Nx2N PUs. *Id.* Lin explains that a block of pixels can be divided into a first PU

("PU1") and second PU ("PU2"). ASUS-1006, ¶25, ¶44, Figs. 7A-7D; ASUS-1013, 000007, 000010, 000017; ASUS-1003, ¶CA170.

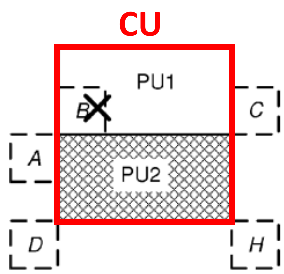


Fig. 7A

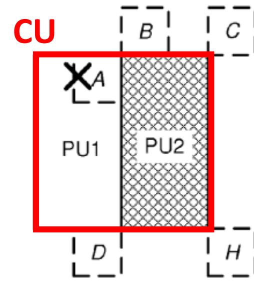


Fig. 7B

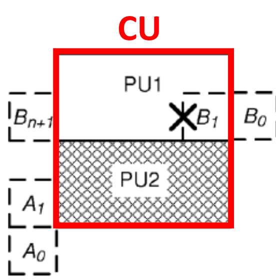


Fig. 7C

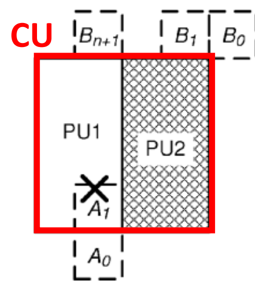


Fig. 7D

Lin examines whether the block is divided into two PUs horizontally (having 2NxN PUs) or vertically (Nx2N). Lin "identifies and removes redundant candidates" by examining the CU for "scenario[s] that… may cause the current PU to be… considered redundant and can be removed[,]" including where "for the second 2NxN,… Nx2N… PU, one or more of the MVP candidates are redundant and removed if said one or more of the MVP candidates located within the previous (first) 2NxN,… Nx2N… PU." ASUS-1006, ¶25, ¶44; ASUS-1013, 000009-11, 000017; ASUS-1003, ¶CA171.

And if so, excluding the potential spatial motion vector prediction candidate from the motion vector prediction list if the prediction unit is the second prediction unit. Lin examines whether (i) the current block is divided into two PUs and (ii) the spatial motion vector prediction candidate is from the other PU; if so, Lin excludes the candidate from the MVP list because it is redundant. ASUS-1006, ¶44, ¶25, Figs. 7A-7D; ASUS-1013, 000010, 000017.

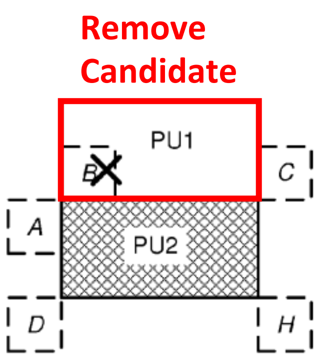


Fig. 7A

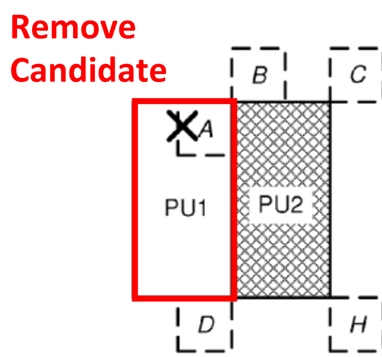


Fig. 7B

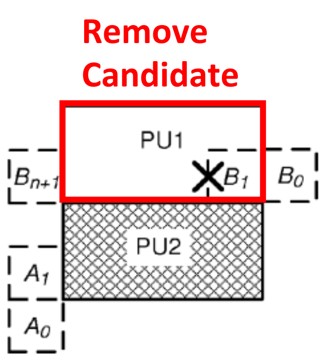


Fig. 7C

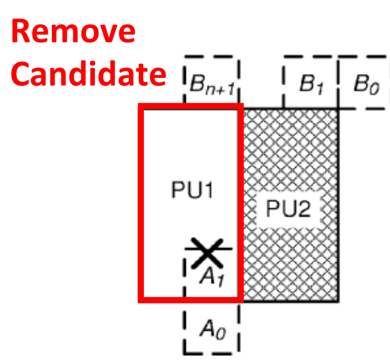


Fig. 7D

ASUS-1003, ¶¶CA172-173.

Motivation to Combine Lin with Rusert/Karczewicz. Furthermore, this concept would have been obvious because a CU comprises one PU when it has uniform motion and is only divided if different parts are moving in different directions, as Karczewicz explains. *See* ASUS-1006, ¶4; ASUS-1013, 000007; ASUS-1005, ¶35. Therefore, for a divided block, the motion vector for one PU is not a good predictor for the other. Lin teaches what Karczewicz suggests, that the motion vector from one half can be removed as a candidate for the other, thereby reducing the number of candidates. ASUS-1006, ¶25, ¶44; ASUS-1013, 000010; ASUS-1003, ¶¶CA174-175. This would have had the predictable result of excluding such candidates, consistent with the reason why a block was divided to begin with. ASUS-1003, ¶CA94.

Accordingly, a POSITA would have been motivated to apply Lin's teachings to the Rusert/Karczewicz combination because Lin provides a straightforward teaching for reducing redundant candidates when blocks are divided. This would have furthered Rusert's goal of "reduc[ing] the number of previous motion vectors that must be considered[.]" ASUS-1004, ¶12, ¶7; ASUS-1003, ¶¶CA88-89. Ground 2 combines Lin's known technique to improve similar H.265 PU-based methods in the same way, which were ready for improvement to reduce the number of previous motion vectors that must be considered. ASUS-1004, ¶12; ASUS-1003, ¶¶CA90-91.

The combination would not have changed the principle of operation for any reference, in the manner taught by each reference. ASUS-1006, ¶44; ASUS-1013, 000010. Ground 2 simply applies Lin's exclusion of PUs in certain scenarios. ASUS-1003, ¶CA95.

Reasonable Expectation of Success. As with Ground 1, all references teach aspects of block-based video encoding for H.264/H.265 to reduce PMV candidates. ASUS-1004, ¶25, ¶39, ¶¶42-44; ASUS-1005, ¶3, ¶5, ¶38; ASUS-1006, ¶4, ¶25; ASUS-1013, 000009-11. The combination does not modify Rusert, Karczewicz, or Lin in a way that would render any reference inoperative. Indeed, many of the same reasons explained for Ground 1 likewise apply to Ground 2. ASUS-1003, ¶¶CA96-98.

To the extent PO argues that Lin provides teachings in the context of merge mode (a mode of operation in H.265), that is irrelevant because the challenged claims do not require or exclude merge mode, and the underlying rationale for Lin's teachings is broadly applicable to scenarios when a block has been divided. With or without merge mode, blocks are divided into multiple PUs to assign them different motion vectors. ASUS-1005, ¶35; ASUS-1006, ¶4; ASUS-1013, 000007; ASUS-1003, ¶CA93.

6. Claim 5

[5pre] The method according to claim 1, further comprising

Ground 1 teaches [5pre]. *Supra* §VI.A.2[1pre]; ASUS-1003, ¶CA178.

[5a]	**determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and**

Ground 1 teaches [5a]. ASUS-1003, ¶¶CA179-183. As explained above, Ground 1 selects spatial motion vector prediction candidates (PMV candidates) as potential candidates to be included in a motion vector prediction list (PMV_CANDS). *Supra* §VI.A.2[1a], §VI.A.3.

Rusert teaches "the number of candidates in PMV_CANDS *may be limited* to a pre-defined or dynamically obtained number" including one, four, or seven. ASUS-1004, ¶73, ¶77, ¶¶84-90, ¶107. By determining this maximum number, Rusert "reduce[s] the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words" and balances the increased chance of a suitable match with the increased cost of longer code words. ASUS-1004, ¶84, ¶13, ¶70, ¶107; ASUS-1003, ¶¶CA180-181.

Additionally, Rusert teaches VLC index values that "depend on the maximum number of candidates in PMV_[C]ANDS" denoted as "Maximum list size C[.]" ASUS-1004, ¶88. Therefore, Rusert determines a maximum number of candidates "C" which dictates index values. ASUS-1003, ¶CA182.

[5b]	**limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.**

Ground 1 teaches [5b]. ASUS-1003, ¶¶CA184-188. 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[.]" ASUS-1004, ¶¶44-48. Furthermore, Rusert teaches "the candidate at the end of the PMV_CANDS list may be removed" to limit "the number of candidates in PMV_CANDS… to a pre-defined… number[.]" ASUS-1004, ¶73. Thus, the number of PMV candidates in PMV_CANDS is limited to be smaller or equal to the pre-defined maximum number. ASUS-1003, ¶¶CA185-186.

Additionally, Rusert teaches VLC examples based on the "Maximum list size C" of PMV_CANDS that limit the number of candidates in PMV_CANDS to be smaller or equal to the maximum number "C". ASUS-1004, ¶88; ASUS-1003, ¶CA187.

7. Claim 6

[6pre] The method according to claim 5 comprising:

Ground 1 teaches [6pre]. *Supra* §VI.A.6; ASUS-1003, ¶¶CA189-190.

[6a]	**examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;**

Ground 1 teaches [6a]. ASUS-1003, ¶¶CA191-194. Rusert continues scanning for new candidates while their number is smaller than a pre-determined number; Rusert "terminat[es]" the scan for new candidates "as soon as a pre-defined number of unique PMV candidates have been found[.]" ASUS-1004, ¶¶44-48, ¶13, ¶¶70-77, ¶¶84-90, ¶107; *supra* §VI.A.6. Additionally, Rusert teaches that "the candidate at the end of the PMV_CANDS list may be removed" to limit "the number of candidates in PMV_CANDS… to a pre-defined… number." ASUS-1004, ¶73; ASUS-1003, ¶¶CA192-193.

[6b]	**if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;**

Ground 1 teaches [6b]. ASUS-1003, ¶¶CA195-204. Rusert obtains potential spatial motion vector prediction candidates from neighboring blocks. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n; *supra* §VI.A.2[1a]. While scanning blocks for new candidates, Rusert examines whether the block to which a potential candidate belongs is available for motion prediction. First, Rusert teaches that blocks "coded after the present block" "would never be available[.]" ASUS-1004, ¶54. Second, blocks are "sometimes… available depending upon the traversal pattern used" to reach the current block, where some blocks would not be coded yet and would not have motion information for motion prediction. ASUS-1004, ¶54. Third, Rusert examines whether blocks have "no motion vector present" or have "the same [motion vector] as a block earlier in the sequence" and are thus not available (since Rusert seeks to avoid duplication). ASUS-1004, ¶54; ASUS-1003, ¶¶CA196-200.

Rusert's block is a PU because it is the unit for which a motion vector is assigned for motion prediction. ASUS-1004, ¶¶2-5, ¶36, ¶43; *supra* §VI.A.2[1a] (explaining PUs). Ground 1 applies Rusert's motion vector teachings to PUs, as explained above. *Supra* §VI.A.1, §VI.A.2[1a]; ASUS-1004, ¶¶3-4, ¶36; ASUS-1005, ¶¶33-36, ¶¶64-66. It would have been obvious to examine whether a PU, which Karczewicz teaches is the block for which motion vectors are assigned, is available for motion prediction because Rusert teaches three reasons why a motion vector might not be available for a block. ASUS-1005, ¶¶33-36, ¶¶64-66; ASUS-1003, ¶¶CA201-203.

[6c]	**if so, performing at least one of the**
…	
[6k]	**…**

Limitation [6c] is satisfied if [6k] is satisfied. ASUS-1003, ¶¶CA205-212.

As explained above, Rusert's block is a PU, and Ground 1 applies Rusert's teachings, e.g. for scanning and analyzing motion vectors from blocks, to PUs. *Supra* §VI.A.2[1a] (explaining Rusert's blocks), §VI.A.1 (motivation to combine); ASUS-1004, ¶¶2-5, ¶36, ¶43; ASUS-1005, ¶¶33-36, ¶¶64-66; ASUS-1003, ¶¶CA205-206.

If Rusert's scan continues (*supra* §VI.A.7[6a]), Ground 1 applies the below-explained teachings **for a potential spatial motion vector prediction candidate** (from block 5), **which is below the potential spatial motion vector prediction candidate on the left side** (from block 2) **of the prediction unit**. Rusert's scan pattern in Fig. 3n proceeds in numerical order. ASUS-1004, ¶¶65-66. The PMV candidate for block 5 is below the candidate for block 2, which is on the left side of the current block (".") for which candidates are being evaluated. ASUS-1004, ¶¶65-66, Fig. 3n:

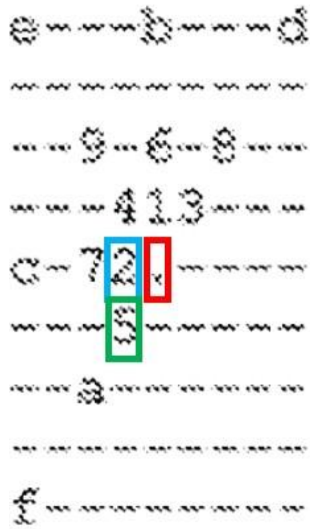


Fig. 3n

These blocks are PUs, and Ground 1 applies these teachings to PUs. ASUS-1003, ¶¶CA207-208.

Excluding the potential spatial motion vector prediction candidate from the motion vector prediction 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. When analyzing a potential PMV 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]." ASUS-1004, ¶¶71-72, ¶21, ¶62. Rusert excludes the potential spatial motion vector prediction candidate from PMV_CANDS if "the PMV candidate is a *duplicate* of another PMV candidate" in PMV_CANDS or if it has essentially similar motion information, e.g., if "the PMV candidate is… within a threshold distance of an existing PMV candidate" in PMV_CANDS. ASUS-1004, ¶21, ¶62, ¶¶71-72. In Fig. 3n, when Rusert evaluates whether to update PMV_CANDS with the candidate from block 5, Rusert compares it with candidates already in PMV_CANDS, e.g., from blocks 1, 2, 3, and 4. ASUS-1004, ¶44, ¶¶65-66, ¶¶71-72, Fig. 3n. If the candidate from block 5 has essentially the same motion information compared with the candidate from block 2, then Rusert excludes the potential new candidate from block 5. *Id.*; ASUS-1003, ¶¶CA209-210.

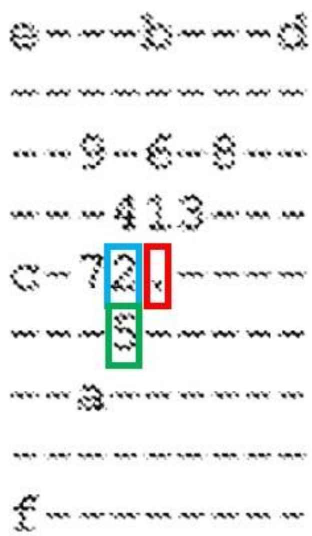


Fig. 3n

Rusert looks to whether the candidates are essentially similar by looking at whether they are duplicates or within a threshold distance. ASUS-1004, ¶72 (excluding "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…"); ASUS-1003, ¶CA211.

8. Claim 7

7.	**The method according to claim 1 further comprising including a temporal motion prediction candidate into the motion vector prediction list.**

Ground 1 teaches claim 7. ASUS-1003, ¶¶CA213-216. "Motion vectors to be added to a PMV_CANDS list may comprise spatial or *temporal neighbors of the current block,* or combinations of spatial and/or temporal neighbors[.]" ASUS-1004, ¶67. Temporal neighbors are "co-located blocks in neighboring frames." ASUS-1004, ¶5. Therefore, Rusert includes a candidate motion vector obtained from a previously-encoded frame (a temporal motion prediction candidate). *Supra* §V.B; ASUS-1003, ¶CA215.

9. Claim 8

8.	**The method according to claim 1 further comprising selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the block of pixels.**

Ground 1 teaches claim 8. ASUS-1003, ¶¶CA217-221. Rusert teaches "signaling which PMV [candidate] is used for motion vector prediction[.]" ASUS-

1004, ¶84; *supra* §§VI.A.2[1a], [1e]. "[A] code 'index'… is sent to *select* a particular PMV candidate… from… PMV_CANDS"; the particular PMV candidate represents the PMV used to "reconstruct MV=DMV+PMV" for the current block of pixels. ASUS-1004, ¶36; ASUS-1003, ¶CA219.

Additionally, Rusert teaches exemplary codes to signal the index of the selected candidate from PMV_CANDS. ASUS-1004, ¶¶88-102; ASUS-1003, ¶CA220.

10. Claim 9

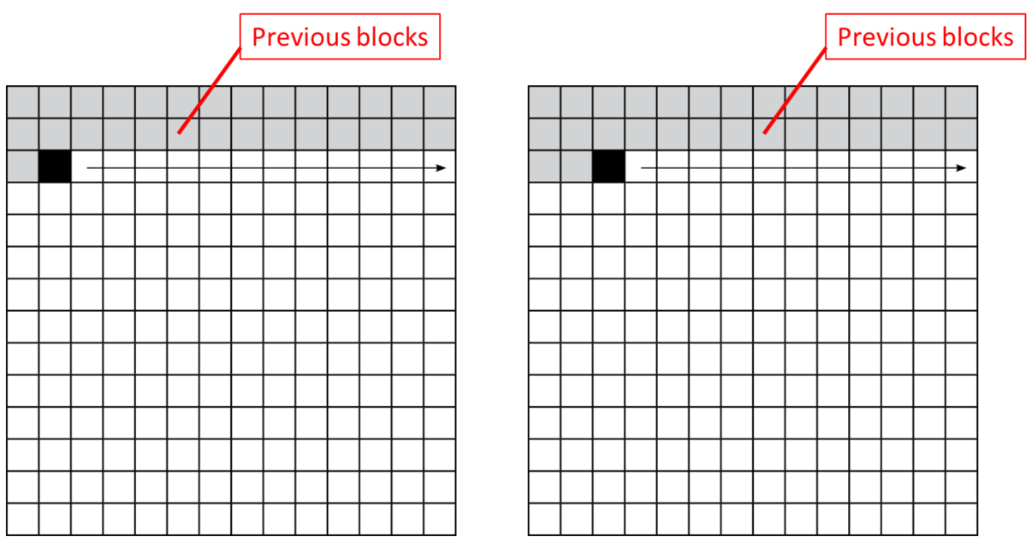
[9pre]	**A method comprising:**

Ground 1 teaches [9pre], e.g., "a *method* of selecting PMV candidates" comprising elements explained below. ASUS-1004, Abstract, ¶1, ¶11; *infra* §§VI.A.10[9a]-[9e]; ASUS-1003, ¶¶CB99-100.

[9a]	**selecting a first spatial motion vector prediction candidate from a set of spatial motion vector prediction candidates for an encoded block of pixels as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list for a prediction unit of the encoded block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;**

Ground 1 teaches [9a]. ASUS-1003, ¶¶CB101-115.

Selecting a first spatial motion vector prediction candidate (PMV candidate) **from a set of spatial motion vector prediction candidates** (set of previously-coded motion vectors) **for an encoded block of pixels**. Rusert teaches "selecting… PMV candidates" for a current block from a "set of previously coded motion vectors that were used for previous blocks." ASUS-1004, ¶¶11-12, ¶15, ¶¶24-25, ¶39, ¶44, ¶¶51-66, ¶113, Fig. 6. As Rusert iterates through blocks of pixels in a frame, each block will have its own unique set of previously-coded motion vectors from which to select a PMV candidate because after a block is encoded/decoded, the set of previously-coded motion vectors increases, as illustrated below. ASUS-1004, ¶2 ("pixel blocks"), ¶¶11-12, ¶36, ¶59, Fig. 3g; ASUS-1003, ¶¶CB102-104:



The decoder applies these teachings to an encoded block of pixels. As Rusert explains, the decoder receives encoded blocks from the encoder (ASUS-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 receives the current block for which candidates are being evaluated. ASUS-1004, ¶35, ¶¶24-25; ASUS-1003, ¶CB104.

Rusert's predicted motion vector ("PMV") (ASUS-1004, ¶3) candidates comprise spatial motion vector prediction candidates obtained from previously-coded blocks in the current frame. *Supra* §V.A. They include "spatially neighboring motion vectors" (ASUS-1004, ¶6, ¶¶3-5) and are included in PMV_CANDS, which "comprise[s] *spatial* … *neighbors* of the current block" in the current frame. ASUS-1004, ¶67, ¶¶4-6; ASUS-1003, ¶CB103.

Selecting a first… candidate… as a potential spatial motion vector prediction candidate to be included in a motion vector prediction list (PMV_CANDS). Rusert teaches search patterns for "an outwards going scan… around the current block" for selecting PMV candidates to potentially be included in PMV_CANDS. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n (showing numerical scan order around current block "." from block 1 onwards, with directional annotation added for 3n):

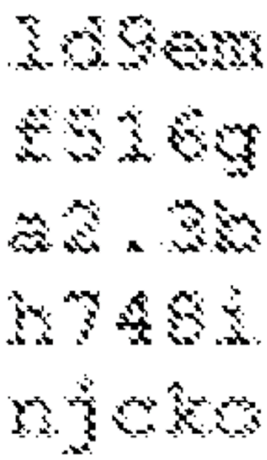


Fig. 3b

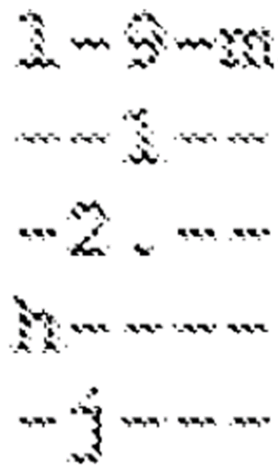


Fig. 3c

Fig. 3n

ASUS-1003, ¶¶CB105-107.

Following these search pattern sequences, Rusert visits a previously-coded block and selects the motion vector for that block as a candidate for potential inclusion in PMV_CANDS. ASUS-1004, ¶44, ¶¶51-66; *infra* §VI.A.10[9c]-[9d] (explaining evaluation for potential inclusion in PMV_CANDS); ASUS-1003, ¶CB107.

PMV_CANDS is **a motion vector prediction list for a prediction unit of the encoded block of pixels**. It is a list of predicted motion vectors "used for coding a motion vector associated with a current… block." ASUS-1004, ¶41, ¶4, ¶¶39-42. PMV_CANDS is "dynamically generated specifically for the current… block[.]" ASUS-1004, ¶41. "[B]efore a block is processed, a PMV_CANDS list is initialized and then updated with… motion vectors." *Id*. When PMV_CANDS is updated, i.e., when “one or more motion vectors are added” (ASUS-1004, ¶39), it comprises a subset of the set of previously-coded motion vectors that have been included in PMV_CANDS to that point. ASUS-1004, ¶¶4-5, ¶¶36-39, ¶¶43-44, ¶¶51-66. When PMV_CANDS is complete, a “predicted motion vector (PMV)” that “is used to predict a [motion vector] is signaled” using an index “to select a particular PMV candidate… from PMV_CANDS[.]” ASUS-1004, ¶36; ASUS-1003,¶CB108.

The prediction unit of Rusert’s encoded block of pixels is the encoded block itself because that is the unit for which a motion vector is assigned for motion prediction. ASUS-1004, ¶¶2-4, ¶43. Rusert provides a motion vector for each “8x8 pixel block,” also called a “sub-block” because it is a portion of a “macroblock.” ASUS-1004, ¶36. Rusert scans neighboring blocks because motion vectors are assigned based on encoded blocks, which are the PUs in Rusert’s teachings. *See* ASUS-1004, ¶¶50-67, Figs. 3a-3n; ASUS-1003, ¶CB109.

Additionally, Ground 1 combines Rusert’s “block” teachings (following H.264 terminology) to PUs (in H.265 terminology). *Supra* §VI.A.1; ASUS-1004, ¶116. As Karczewicz explains, H.265 introduced terminology for a type of block called “prediction unit[s.]” ASUS-1005, ¶¶33-36; *supra* §I. “In general, a CU has a similar purpose to a macroblock of H.264” (ASUS-1005, ¶33), which in its simplest case is commensurate with a PU but may also be divided into multiple

PUs. ASUS-1005, ¶35. "[T]he PU may include data defining a motion vector for the PU" that is used for "prediction using a PU[.]" ASUS-1005, ¶¶35-36, ¶66; ASUS-1003, ¶¶CB110-111. In short, Karczewicz explains that H.265 assigned motion vectors based on a type of block called a PU. ASUS-1005, ¶¶33-36, ¶64. Rusert explained its teachings based on the blocks for which motion vectors were assigned (ASUS-1004, ¶¶2-5, ¶36, ¶43). Both Karczewicz and Rusert teach the encoder and decoder use "reciprocal… techniques[,]" which confirms these teachings are applicable to both encoding and decoding. ASUS-1005, ¶50; ASUS-1004, ¶35, ¶¶24-25. Therefore, it would have been obvious to apply Rusert's teachings to PUs, with PMV_CANDS being a motion vector prediction list for a PU of the encoded block of pixels. ASUS-1005, ¶¶33-36, ¶66; ASUS-1003, ¶CB110-112.

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[.]" ASUS-1004, ¶35, ¶¶24-25; ASUS-1003, ¶CB112.

Where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates. PMV_CANDS is a "list of PMV candidates." ASUS-1004, ¶37. Each candidate is a motion vector (ASUS-1004, ¶11, ¶¶24-25, ¶¶39-41) including "x and y components" (ASUS-1004, ¶106,

¶36, ¶¶91-94, ¶100). Therefore, Rusert's PMV_CANDS includes motion information, including motion vectors and their x and y components, of the PMV candidates. ASUS-1004, ¶¶2-3; ASUS-1003, ¶CB113.

Additionally, Karczewicz teaches "the 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[.]" ASUS-1005, ¶35. It would have been obvious to include this PU information in PMV_CANDS because the combination relies on Karczewicz's PU teachings. *Supra* §VI.A.1; ASUS-1003, ¶¶CB114-115.

[9b]	**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;**

Ground 1 teaches [9b]. ASUS-1003, ¶¶CB116-128.

Determining a subset of spatial motion vector prediction candidates (e.g., based on Rusert's scan of previous PMV candidates within an allowed distance and pre-defined number). Rusert "select[s] a set of PMV candidates as a *subset* of the set of previously coded motion vectors that were used for previous blocks[.]" ASUS-1004, ¶¶11-12, ¶¶24-25; *supra* §V.A. "[T]he 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[.]"

ASUS-1004, ¶15, ¶37, Figs. 2a-2b; ASUS-1003, ¶¶CB117-118.

Rusert implements the subset of PMV candidates using an outward scan of blocks having previously-coded motion vectors, starting from the current block and staying within an allowed distance. ASUS-1004, ¶¶4-5, ¶¶36-39, ¶¶43-44, ¶¶51-66. As the scan progresses outwards, the subset includes the previously-coded PMV candidates obtained from blocks in previous locations of Rusert's scan. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n. For example, when the selected first spatial motion vector prediction candidate is obtained from the third block in Rusert's scan sequence, the subset of spatial motion vector prediction candidates comprises the candidates from previous scan locations (the first and second blocks). *Id.*

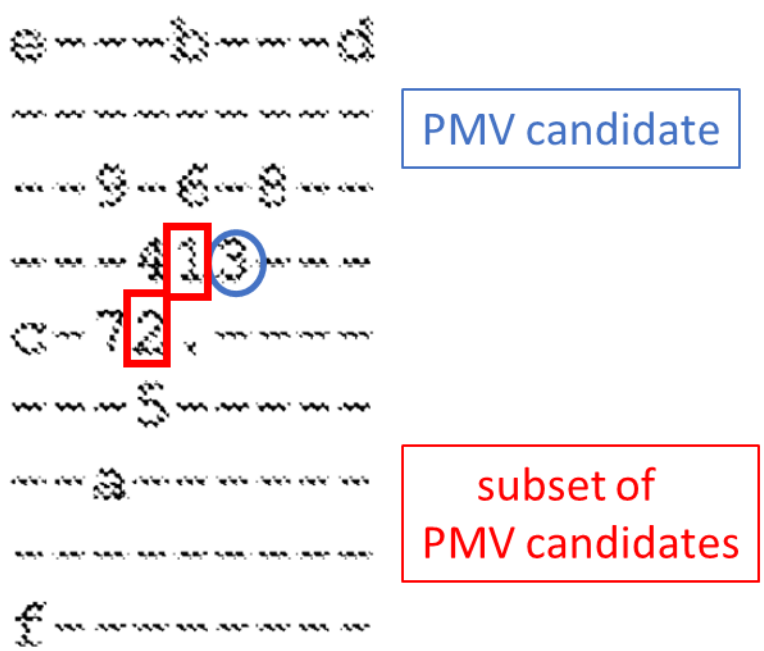


Fig. 3n

ASUS-1003, ¶¶CB119-120.

Additionally, Rusert terminates the scan "as soon as a pre-defined number of unique PMV candidates have been found." ASUS-1004, ¶48. Even PMV candidates that are within "a certain distance" and part of "a predetermined scan pattern" are not considered if the "pre-defined number of unique PMV candidates have been found." ASUS-1004, ¶¶44-49. Rusert improves coding efficiency by using subsets. ASUS-1004, ¶¶12-13. Therefore, Rusert's scan of previously-coded motion vectors is further cabined into a subset because it is limited to this predefined number of PMV candidates. Rusert does not reach every candidate in every block of its scan sequence and instead determines a subset of PMV candidates, from the scan sequence, within a pre-defined number. 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., 7), without using the remaining candidates in the scan sequence. ASUS-1004, ¶48, ¶107; ASUS-1003, ¶¶CB120-121.

As the scan progresses outwards and the subset of PMV candidates is updated, the subset is stored as a list of PMV candidates called PMV_CANDS. ASUS-1004, ¶¶37-41, ¶¶44-49, ¶¶51-66; *infra* §VI.A.10[9d]. The subset is determined for reducing the number of candidates, as explained for [9c]. ASUS-1003, ¶¶CB120.

Based on a location of the block associated with the first spatial motion vector prediction candidate.

Under the interpretation that "the" block is the block from which the selected first candidate is obtained, Rusert teaches this limitation. *Supra* §V.C. A "position" of a block of a PMV candidate is represented as (xpos, ypos), i.e., a location of the block associated with the first spatial motion vector prediction candidate. ASUS-1004, ¶¶51-52. Based on the block's location, the subset of PMV candidates with which the PMV candidate is compared includes the candidates of blocks located in the scan pattern up to the PMV candidate. ASUS-1004, ¶44. For example, in Fig. 3n, the PMV candidate for block "3" is compared with a subset of PMV candidates for blocks "1" and "2" for de-duplication, as explained for [9c]-[9d]. ASUS-1004, ¶44, ¶¶65-66, Fig. 3n:

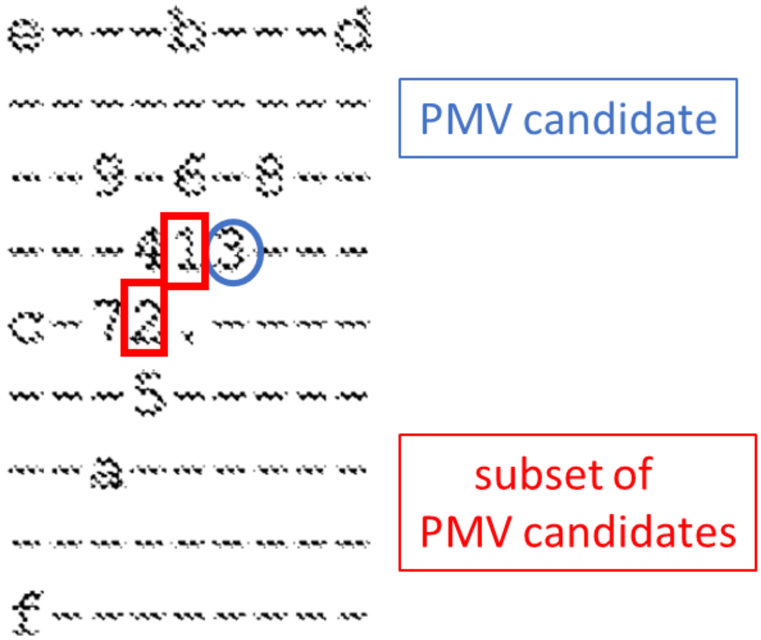


Fig. 3n

ASUS-1003, ¶CB127.

Even if "the block" refers to the "block of pixels" in [9a] and therefore refers to the current block for which candidates are being analyzed, Rusert teaches this limitation. *Supra* §V.C. Because the subset of PMV candidates is based on "an outwards going scan… around the current block" (ASUS-1004, ¶¶43-44), the subset of blocks that are scanned and the corresponding subset of spatial motion vector prediction candidates from those blocks are based on the location of the current block. ASUS-1004, ¶44; ASUS-1003, ¶¶CB122-123.

Furthermore, the subset of PMV candidates is based on "an allowed distance from the current block and an allowed position." ASUS-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. ASUS-1004, ¶¶11-17, ¶¶24-25, ¶¶43-44, ¶¶51-66, Figs. 3a-3n. Rusert teaches exemplary distance values[11] based on the location of the current block ("."). ASUS-1004, ¶47, ¶13, ¶43, Figs. 3a, 3d, 3f, 3g:

Fig. 3a

Fig. 3d

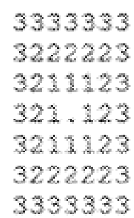


Fig. 3f

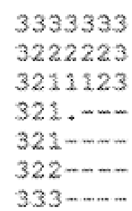


Fig. 3g

Only candidates from blocks with allowed positions relative to the current block are included. ASUS-1004, ¶¶15-16, ¶59 (excluding "blocks to the right and below the current block"), ¶65; ASUS-1003, ¶¶CB124-126.

[11] "Euclidean distance" is occasionally misspelled "Euclidian distance." ASUS-1004, ¶44, ¶78, ¶87; ASUS-1003, ¶CA124n.3.

[9c]	**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;**

Ground 1 teaches [9c]. ASUS-1003, ¶¶CB129-141.

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. Rusert teaches three comparisons for excluding "unnecessary" candidates; each satisfies [9c]. ASUS-1004, ¶21, ¶¶71-72; *supra* §VI.A.10[9b]; ASUS-1003, ¶CB130.

First, Rusert compares whether "the PMV candidate is a duplicate of *another* PMV candidate" in the subset of PMV candidates, which is stored as a list in PMV_CANDS. ASUS-1004, ¶21, ¶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" thus "reducing the number of candidates[.]" ASUS-1004, ¶71, ¶62. A potential PMV candidate is selected and compared with each PMV candidate in the subset of preceding PMV candidates in Rusert's scan sequence, which are the candidates already stored in PMV_CANDS. *Supra* §VI.A.10[9b]. If the x and y components of two motion vectors are the same, they are duplicates and the potential candidate is excluded. *Id.*; ASUS-1003, ¶CB131.

Second, Rusert compares whether "the PMV candidate is… within a threshold distance of an existing PMV candidate" in the subset of candidates stored in PMV_CANDS, e.g., using "a similarity measure" such as Euclidean distance to calculate whether the difference between the x and y components of their motion vectors is "smaller than a pre-defined threshold"; if so, the potential candidate is removed/skipped. ASUS-1004, ¶21, ¶72, ¶87. The distance between duplicate PMV candidates is zero. ASUS-1003, ¶CB132, ¶CB132n.4.

Third, Rusert compares whether "at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits" (ASUS-1004, ¶21), thereby "removing PMV candidates" that "will never be used" because another candidate in the subset stored in PMV_CANDS facilitates "a bit sequence that is shorter or of the same length compared for all possible motion vectors." ASUS-1004, ¶¶90-94. When the x and y values of an existing candidate's motion vector can be added to the x and y values of a difference motion vector ("DMV")[12] to yield the x and y values of the selected candidate while using fewer overall bits, Rusert excludes the selected candidate. ASUS-1004, ¶95 (teaching exemplary PMV/DMV

[12] *Supra* §I.

combinations requiring fewer bits); ASUS-1003, ¶¶CB133-134.

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 Rusert improves efficiency by determining this subset and comparing potential new candidates to the subset. ASUS-1004, ¶¶88-98; ASUS-1003, ¶CB135.

All three teachings compare motion information of the PMV candidates, including the x and y components of their motion vectors. This is motion information of the PMV candidates because they describe the "motion of pixel blocks across frames[.]" ASUS-1004, ¶2. Motion vector coding includes an MVP combined with a DMV. ASUS-1004, ¶3, ¶¶36-37; *supra* §VI.A.10[9a] (explaining motion information); ASUS-1003, ¶CB135.

Rusert teaches or at least suggests performing three comparisons when evaluating whether to update PMV_CANDS with a PMV candidate. ASUS-1004, ¶71, ¶75, ¶¶84-87, ¶90. This process is used when decoding an encoded block of pixels; the same list of candidates is constructed for encoding and decoding. ASUS-1004, ¶35; ASUS-1003, ¶CB136.

Comparing … without making a comparison of each possible candidate pair from the set of spatial motion vector prediction candidates. Each of

Rusert's three teachings (explained above) compares motion information of a selected candidate with the subset of PMV candidates, which is stored in PMV_CANDS, without making a comparison of each possible pair from the set of motion vectors used for all previous blocks. ASUS-1004, ¶¶11-13, ¶¶15-16, ¶113. Rusert teaches that the subset of PMV candidates is indeed a "*subset* of the motion vectors previously used for previous blocks" meaning Rusert does not compare each possible pair from the "set of previously coded motion vectors… used for previous blocks[.]" ASUS-1004, ¶¶11-12, ¶¶24-25. Even within the scan order, Rusert compares potential candidates with preceding candidates already in PMV_CANDS, which are a subset of the candidates in the scan order; therefore, Rusert avoids comparing each possible pair of candidates from the scan order. *See id.*; ASUS-1003, ¶CB137.

Moreover, even within Rusert's scan patterns, Rusert teaches that a "scan may be terminated… as soon as a pre-defined number of unique PMV candidates have been found[.]" ASUS-1004, ¶48. For example, Rusert teaches "us[ing] a maximum of four candidates in the [PMV_CANDS] list" or "us[ing] seven" as the maximum. ASUS-1004, ¶107. Therefore, Rusert does not compare each possible pair of candidates from the set of all previously-coded blocks, or even each possible pair of blocks within the scan order. Rusert thus teaches Claim 9 with the recited "set" being either (a) the previously-coded motion vectors for that frame, 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. ASUS-1003, ¶CB140.

Additionally, because the subset is limited to candidates having "an allowed distance" and "position[,]" candidates from previous blocks outside the allowed distance/position are not in PMV_CANDS and, therefore, are not compared. ASUS-1004, ¶15; *supra* §VI.A.10[9b]. Furthermore, Rusert teaches scan patterns that exclude certain blocks within the allowed distance. ASUS-1004, ¶¶64-65, Figs. 3m, 3n. Given these excluded blocks, there is no comparison of each pair of motion vectors from the set of all previously-coded blocks. ASUS-1003, ¶¶CB138-139.

[9d]	**determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and**

Ground 1 teaches [9d]. ASUS-1003, ¶¶CB142-146.

As explained above, Rusert determines whether to include or exclude the selected first PMV candidate in PMV_CANDS, which stores the subset of PMV candidates, based on each of the three comparisons explained for [9c]. Rusert removes "unnecessary PMV candidates" to "ensure[] the length of the

[PMV_CANDS] list is not unnecessarily long, which would reduce coding efficiency." ASUS-1004, ¶¶20-21. Rusert explains that "[a] PMV candidate *may be determined to be unnecessary*" and be therefore excluded if at least one of the three comparisons is fulfilled. ASUS-1004, ¶21; *supra* §VI.A.10[9c]. "Unnecessary PMV candidates are removed… because it may happen that some candidates in the list will never be used." ASUS-1004, ¶90. A shorter PMV_CANDS "allows the remaining PMV candidates to be signaled using shorter codes and so fewer bits[.]" ASUS-1004, ¶22, ¶90; ASUS-1003, ¶¶CB143-144.

Rusert excludes unnecessary candidates when deciding whether to add a new candidate to the subset of PMV candidates stored in PMV_CANDS. ASUS-1004, ¶21, ¶¶71-72; s*upra* §VI.A.10[9c]. As part of the "update" process whereby new candidates are added, Rusert determines whether a selected PMV candidate should be included or skipped, meaning it is excluded based on any of the three comparisons from [9c]. ASUS-1004, ¶¶71-72. Additionally, this would have been obvious because Rusert teaches the advantages of reducing the number of candidates in PMV_CANDS using the three comparisons from [9c] (ASUS-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 subset of PMV 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[.]” ASUS-1004, ¶90; ASUS-1004, ¶145.

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

Ground 1 teaches [9e]. ASUS-1003, ¶¶CB147-152.

Selecting a spatial motion vector prediction candidate from the motion vector prediction list for use in decoding the encoded block of pixels. Rusert teaches “a code ‘index’… to select a particular PMV candidate… from… PMV_CANDS” and “[u]sing the transmitted index” to “determine the PMV” and reconstruct the motion vector. ASUS-1004, ¶¶36-37. These techniques are “for video decoding… the current block[.]” ASUS-1004, ¶23, ¶35. The current block is in an “encoded video stream” that is “passe[d] to a decoder… employed in decoding the encoded video stream.” ASUS-1004, ¶34; ASUS-1003, ¶CB148.

The spatial motion vector prediction candidate is selected from the motion vector prediction list using information that was received identifying a respective spatial motion vector prediction candidate from the motion vector prediction list constructed by an encoder. Rusert teaches “a particular PMV candidate" is selected "from a list of PMV candidates, PMV_CANDS[,]" using "a code 'index'" sent "from the encoder" and received by the decoder. ASUS-1004, ¶¶36-37, Fig. 1:

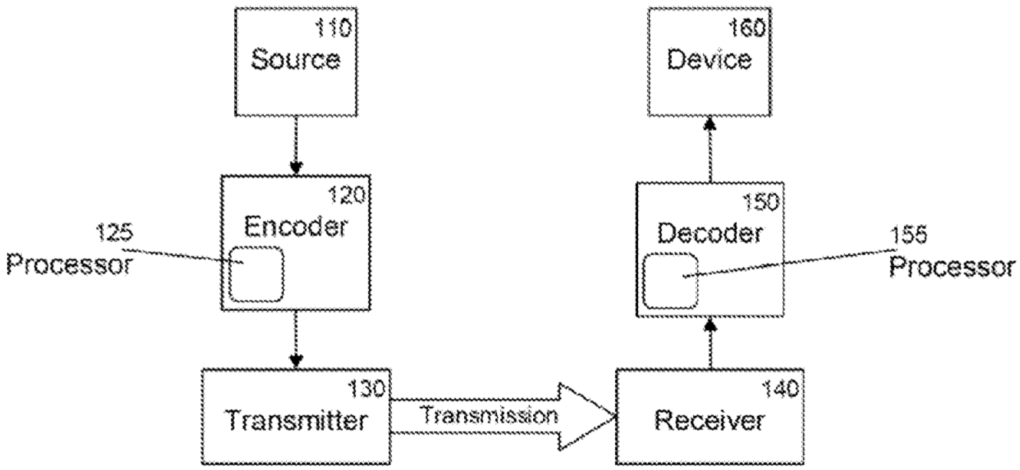


Fig. 1

The received index is information that identifies a respective PMV candidate from PMV_CANDS by pointing to a particular candidate based on its position in PMV_CANDS. In this way, it is an index into PMV_CANDS. Rusert teaches zero-based indexing examples, starting with zero for the first candidate and ending with n-1 for a list of n candidates. ASUS-1004, ¶¶88-95; ASUS-1003, ¶CB149.

PMV_CANDS is constructed by an encoder, and the decoder "mimics the encoder" to construct the same PMV_CANDS. ASUS-1004, ¶¶35-39. From PMV_CANDS, the decoder uses the index "signaled from the encoder" to "select a particular PMV candidate[.]" ASUS-1004, ¶36. "Using the transmitted index, the decoder… can determine the PMV 220" to "reconstruct [the motion vector.]"

ASUS-1004, ¶27. Rusert illustrates this below. ASUS-1004, Figs 2a-2b:

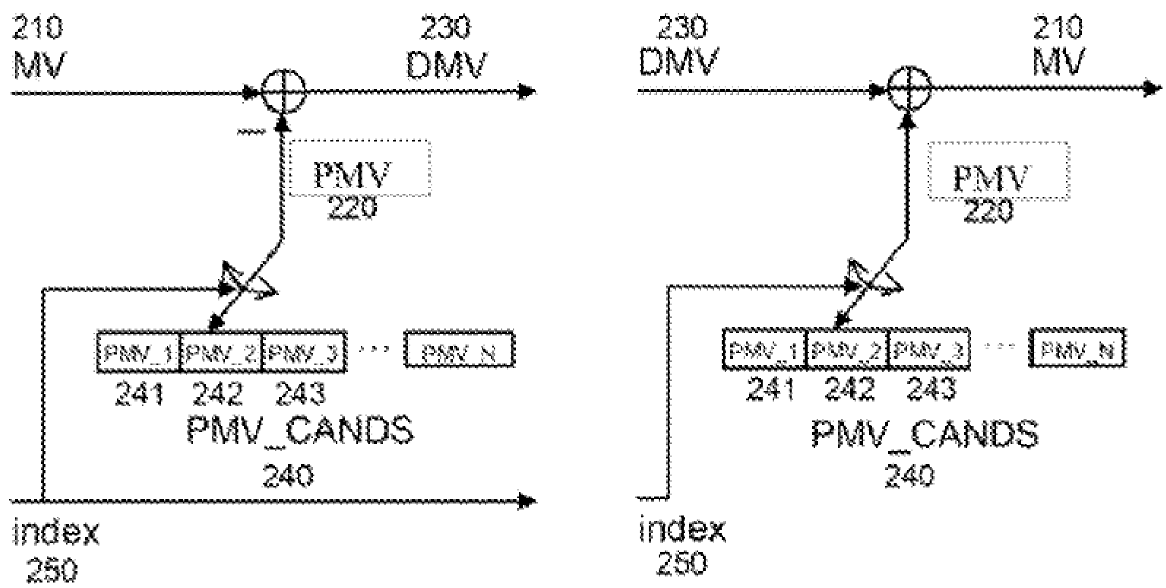


Fig. 2a

Fig. 2b

ASUS-1003, ¶¶CB150-151.

11. Claim 10

10.	**The method according to claim 9 further 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.**

Ground 1 teaches claim 10. ASUS-1003, ¶¶CB153-157. Rusert teaches three comparisons; each satisfies the comparing step explained for [9c]. *Supra* §VI.A.10[9c]; ASUS-1004, ¶21. Rusert further teaches claim 3's additional limitation of comparing motion information for **at most one other** candidate of the set. For Rusert's scan sequences (*supra* §§VI.A.10[9a]-[9b]; ASUS-1004, ¶44,

¶¶51-66, Figs. 3a-3n), the "PMV_CANDS list may be initialized e.g. as an empty list (zero entries)" and then "updated to include previously coded motion vectors[.]" ASUS-1004, ¶39, ¶44, ¶71; *supra* §VI.A.10[9c]. When the scan moves to update PMV_CANDS with the second PMV candidate in the sequence, that candidate is "compar[ed with] the candidates already in the list[,]" which is at most the first PMV candidate in the scan sequence. ASUS-1004, ¶21, ¶¶38-40, ¶44, ¶71. Therefore, the motion information of the second candidate in the sequence is compared with at most one other PMV candidate (the first PMV candidate). ASUS-1003, ¶¶CB154-156.

12. Claim 11

11.	**The method according to claim 9 further 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 motion vector prediction list if the prediction unit is the second prediction unit.**

Rusert and Karczewicz teach **the method according to claim 9**. A POSITA would have been motivated to further apply Lin's teachings, which satisfy claim 11. ASUS-1003, ¶¶CB88-98, ¶¶CB158-165.

Examining whether the received encoded block of pixels is divided into a first prediction unit and a second prediction unit. Rusert teaches receiving an encoded block of pixels (the current block). ASUS-1004, ¶¶34-36, Fig. 1; *supra*

§§VI.A.10[9a], [9e]; ASUS-1003, ¶CB159.

Lin examines whether such blocks are divided. As Lin explains, in H.265/HEVC, "[t]he basic unit for compression, termed Coding Unit (CU), is a 2Nx2N square block, and… [e]ach CU contains one or multiple Prediction Units (PUs)" with divisions "correspond[ing] to horizontal and vertical partition[s.]" ASUS-1006, ¶4; ASUS-1013, 000007, 000017. Horizontally-divided CUs have PUs of size 2NxN; vertically-divided CUs have Nx2N PUs. *Id.* Lin explains that a block of pixels can be divided into a first PU ("PU1") and a second PU ("PU2"). ASUS-1006, ¶6, ¶25, ¶44, Figs. 7A-7D; ASUS-1013, 000007, 000010, 000017; ASUS-1003, ¶CB159.

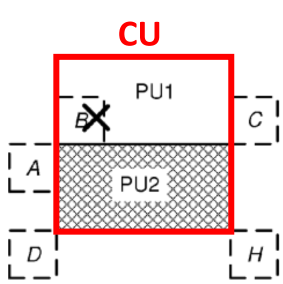


Fig. 7A

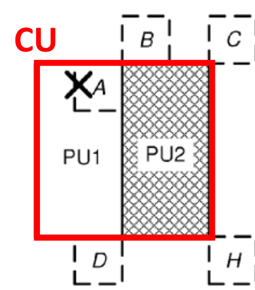


Fig. 7B

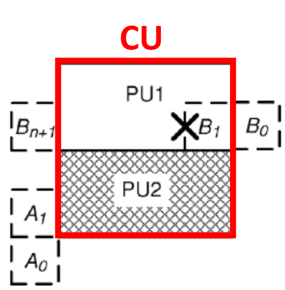


Fig. 7C

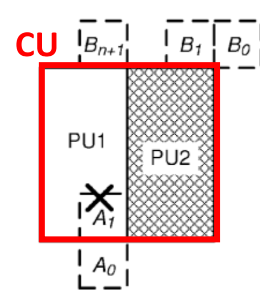


Fig. 7D

Lin examines whether the received encoded block is divided into two PUs horizontally (having 2NxN PUs) or vertically (Nx2N). Lin "identifies and removes redundant candidates" by examining the CU for "scenario[s] that… may cause the current PU to be… considered redundant and can be removed[,]" including where "for the second 2NxN,… Nx2N… PU, one or more of the MVP candidates are redundant and removed if said one or more of the MVP candidates located within the previous (first) 2NxN,… Nx2N… PU." ASUS-1006, ¶25, ¶44; ASUS-1013, 000009-11, 000017. These teachings apply to an encoded block of pixels received by a decoder, which "mimics the encoder in order to achieve encoder/decoder synchronization." ASUS-1004, ¶35, ¶¶24-35; ASUS-1005, ¶50; ASUS-1006, ¶22, ¶47; ASUS-1013, 000008; ASUS-1003, ¶CB160.

And if so, excluding the potential spatial motion vector prediction candidate from the motion vector prediction list if the prediction unit is the second prediction unit. Lin examines whether (i) the current block is divided into two PUs and (ii) the spatial motion vector prediction candidate is from the other PU; if so, Lin excludes the candidate from the MVP list because it is redundant. ASUS-1006, ¶44, ¶25, Figs. 7A-7D; ASUS-1013, 000010, 000017.

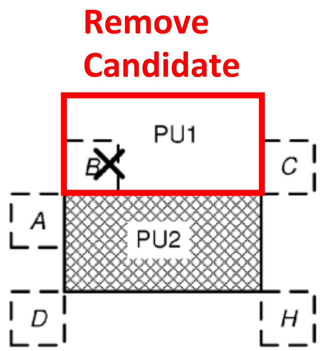


Fig. 7A

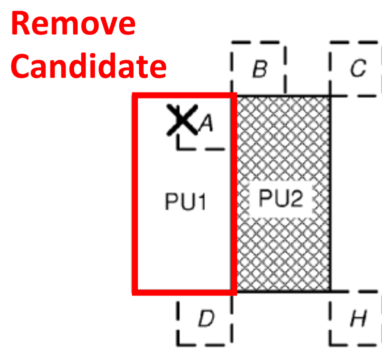


Fig. 7B

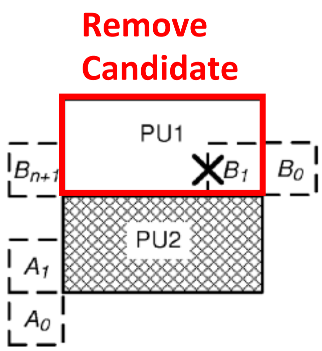


Fig. 7C

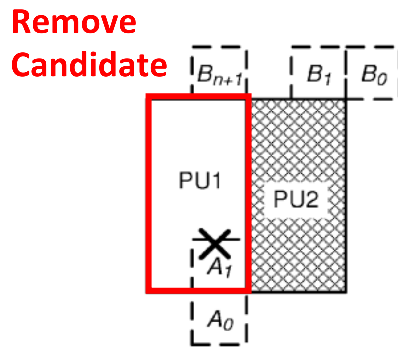


Fig. 7D

ASUS-1003, ¶¶CB161-162.

13. Claim 12

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

Ground 1 teaches [12pre]. *Supra* §VI.A.10[9pre]; ASUS-1003, ¶CB166.

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

Ground 1 teaches [12a]. ASUS-1003, ¶¶CB167-171. As explained above, Ground 1 selects spatial motion vector prediction candidates (PMV candidates) as potential candidates to be included in a motion vector prediction list (PMV_CANDS). *Supra* §VI.A.10[9a]. ASUS-1003, ¶CB167.

Rusert teaches "the number of candidates in PMV_CANDS *may be limited* to a pre-defined or dynamically obtained number" including one, four, or seven. ASUS-1004, ¶73, ¶77, ¶¶84-90, ¶107. By determining this maximum number, Rusert "reduce[s] the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words" and balances the increased chance of a suitable match with the increased cost of longer code words. ASUS-1004, ¶84, ¶13, ¶70, ¶107; ASUS-1003, ¶¶CB168-169.

Additionally, Rusert teaches VLC index values that "depend on the maximum number of candidates in PMV_[C]ANDS" denoted as "Maximum list size C[.]" ASUS-1004, ¶88. Therefore, Rusert determines a maximum number of candidates "C" which dictates index values. ASUS-1003, ¶CB170.

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

Ground 1 teaches [12b]. ASUS-1003, ¶¶CB172-176. 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[.]" ASUS-1004, ¶¶44-48. Furthermore, Rusert teaches "the candidate at the end of the PMV_CANDS list may be removed" to limit "the number of candidates in PMV_CANDS… to a pre-defined… number[.]" ASUS-1004, ¶73. Thus, the number of PMV candidates in PMV_CANDS is limited to be smaller or equal to the pre-defined maximum number. ASUS-1003, ¶¶CB173-174.

Additionally, Rusert teaches VLC examples based on the "Maximum list size C" of PMV_CANDS that limit the number of candidates in PMV_CANDS to be smaller or equal to the maximum number "C". ASUS-1004, ¶88; ASUS-1003, ¶CB175.

14. Claim 13

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

Ground 1 teaches [13pre]. *Supra* §VI.A.13; ASUS-1003, ¶¶CB177-178.

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

Ground 1 teaches [13a]. ASUS-1003, ¶¶CB179-182. Rusert continues scanning for new candidates while their number is smaller than a pre-determined number; Rusert "terminat[es]" the scan for new candidates "as soon as a pre-defined number of unique PMV candidates have been found[.]" ASUS-1004, ¶¶44-

48, ¶13, ¶¶70-77, ¶¶84-90, ¶107; *supra* §VI.A.13. Additionally, Rusert teaches that "the candidate at the end of the PMV_CANDS list may be removed" to limit "the number of candidates in PMV_CANDS… to a pre-defined… number." ASUS-1004, ¶73; ASUS-1003, ¶¶CB180-181.

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

Ground 1 teaches [13b]. ASUS-1003, ¶¶CB183-192. Rusert obtains potential spatial motion vector prediction candidates from neighboring blocks. ASUS-1004, ¶44, ¶¶51-66, Figs. 3a-3n; *supra* §VI.A.10[9a]. While scanning blocks for new candidates, Rusert examines whether the block to which a potential candidate belongs, is available for motion prediction. First, Rusert teaches that blocks "coded after the present block" "would never be available[.]" ASUS-1004, ¶54. Second, blocks are "sometimes… available depending upon the traversal pattern used" to reach the current block, where some blocks would not be coded yet and would not have motion information for motion prediction. *Id*. Third, Rusert examines whether blocks have "no motion vector present" or have "the same [motion vector] as a block earlier in the sequence" and are thus not available (since Rusert seeks to avoid duplication). *Id*.; ASUS-1003, ¶¶CB184-188.

Rusert's block is a prediction unit because it is the unit for which a motion vector is assigned for motion prediction. ASUS-1004, ¶¶2-5, ¶36, ¶43; *supra* §VI.A.10[9a] (explaining PUs). Ground 1 applies Rusert's motion vector teachings to PUs, as explained above. *Supra* §VI.A.1, §VI.A.10[9a]; ASUS-1004, ¶¶3-4, ¶36; ASUS-1005, ¶¶33-36, ¶¶64-66. It would have been obvious to examine whether a PU, which Karczewicz teaches is the block for which motion vectors are assigned, is available for motion prediction because Rusert teaches three reasons why a motion vector might not be available for a block. ASUS-1005, ¶¶33-36, ¶¶64-66; ASUS-1003, ¶¶CB189-191.

[13c]	**if so, performing at least one of the**
…	
[13k]	**…**

Limitation [13c] is satisfied if [13k] is satisfied. ASUS-1003, ¶¶CB193-200.

As explained above, Rusert's block is a prediction unit, and Ground 1 applies Rusert's teachings, e.g. for scanning and analyzing motion vectors from blocks, to PUs. *Supra* §VI.A.10[9a] (explaining Rusert's blocks), §VI.A.1 (motivation to combine); ASUS-1004, ¶¶2-5, ¶36, ¶43; ASUS-1005, ¶¶33-36, ¶¶64-66; ASUS-1003, ¶¶CB193-194.

If Rusert's scan continues (*supra* §VI.A.14[13a]), Ground 1 teaches performing the step of [13k]. Rusert applies the below-explained teaching **for a**

potential spatial motion vector prediction candidate (from block 5)**, which is below the potential spatial motion vector prediction candidate on the left side** (from block 2) **of the prediction unit**. Rusert's scan pattern in Fig. 3n proceeds in numerical order. ASUS-1004, ¶¶65-66. The PMV candidate for block 5 is below the candidate for block 2, which is on the left side of the current block (".") for which candidates are being evaluated. *Id*., Fig. 3n:

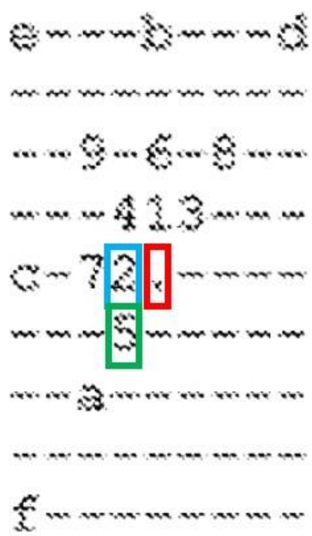


Fig. 3n

These blocks are PUs, and Ground 1 applies these teachings to PUs. ASUS-1003, ¶¶CB195-196.

Excluding the potential spatial motion vector prediction candidate from the motion vector prediction 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. When analyzing a potential PMV 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]." ASUS-1004, ¶¶71-72, ¶21. Rusert excludes the potential candidate from PMV_CANDS if "the PMV candidate is a *duplicate* of another PMV candidate" in PMV_CANDS or if it has essentially similar motion information, e.g., if "the PMV candidate is… within a threshold distance of an existing PMV candidate" in PMV_CANDS. ASUS-1004, ¶21, ¶62, ¶¶71-72, ¶87. In Fig. 3n, when Rusert evaluates whether to update PMV_CANDS with the candidate from block 5, Rusert compares it with candidates already in PMV_CANDS, e.g., from blocks 1, 2, 3, and 4. ASUS-1004, ¶44, ¶¶65-66, ¶¶71-72, Fig. 3n. If the candidate from block 5 has essentially the same motion information compared with the candidate from block 2, then Rusert excludes the potential candidate from block 5. *Id*.; ASUS-1003, ¶¶CB197-198.

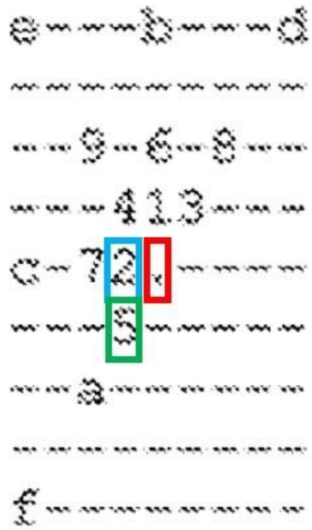


Fig. 3n

Rusert looks to whether the candidates are essentially similar by looking at whether they are duplicates or within a threshold distance. ASUS-1004, ¶72 (excluding "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…"); ASUS-1003, ¶CB199.

15. Claim 14

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

Ground 1 teaches claim 14. ASUS-1003, ¶¶CB201-206. Rusert teaches "signaling which PMV [candidate] is used for motion vector prediction[.]" ASUS-

1004, ¶84; *supra* §§VI.A.10[9a], [9e]. "[A] code 'index'… is sent to *select* a particular PMV candidate… from a list of PMV candidates, PMV_CANDS"; the particular PMV candidate represents the PMV used to "reconstruct MV=DMV+PMV" for the current encoded block of pixels. ASUS-1004, ¶36; ASUS-1003, ¶CB203.

Additionally, Rusert teaches exemplary codes to signal the index of the selected candidate from PMV_CANDS. ASUS-1004, ¶¶88-102. Rusert applies the above teachings to "video decoding, wherein the current block is the block being… decoded" (ASUS-1004, ¶23, ¶35) and is part of an "encoded video stream" that is "passe[d] to a decoder[.]" ASUS-1004, ¶¶34-36; *supra* §VI.A.10[9e]; ASUS-1003, ¶¶CB204-205.

16. Claims 15, 29

Beyond the preambles, claims 15 and 29 are nearly identical to claim 1, with limitations [15a]-[15e]/[29a]-[29e] having only minor differences with [1a]-[1e], respectively. Limitations [15c]/[29c] recite "each possible candidate pair" where [1c] recites "each pair" and is taught by Ground 1 because not comparing each pair satisfies not comparing each possible candidate pair. ASUS-1003, ¶CA234, ¶CA256.

Ground 1 teaches [15a]-[15e]/[29a]-[29e] for the same reasons provided for [1a]-[1e]. *Supra* §VI.A.2[1a]-[1e]; ASUS-1003, ¶¶CA225-241, ¶¶CA252-

258.

[15pre]	**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:**
[29pre]	**A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprising instructions for:**

Ground 1 teaches [15pre] and [29pre]. Rusert teaches a video encoding/decoding "apparatus" (ASUS-1004, ¶1, ¶¶24-27, ¶114, ¶116, claims 17-18) comprising "a processor" and "a computer-readable medium, carrying instructions, which when executed[,]… causes [the processor] to carry out any of the methods disclosed herein." ASUS-1004, ¶¶24-26, ¶1; ASUS-1003, ¶¶CA222-224, ¶¶CA249-251.

17. Claims 16-22

Other than the preambles, claims 16-22 are identical to claims 2-8. Ground 1 teaches the preambles (apparatus of claim 15). *Supra* §VI.A.16. Grounds 1 and 2 teach the rest of claims 16-22 for the reasons provided for claims 2-8, respectively. *Supra* §VI.A.3-9; ASUS-1003, ¶¶CA242-248.

18. Claims 23, 30

Beyond the preambles, claims 23 and 30 are nearly identical to claim 9, with limitations [23a]-[23e]/[30a]-[30e] having only minor differences with [9a]-[9e], respectively, and being satisfied for reasons explained above. *Supra* §§VI.A.10[9a]-[9e]. Limitations [23c]/[30c] recite "the… candidate in the determined subset" where [9c] recites "another… candidate of the set"; Rusert satisfies both by teaching a candidate in the subset (explained for [9b]), which is also another candidate in the larger set of previously-coded motion vectors. Limitation [23d] repeats language from [23c], which Ground 1 teaches as explained for [23c] and [9c]. ASUS-1003, ¶¶CB210-222, ¶¶CB234-239.

[23pre] 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: **[30pre] A non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder, said program codes comprising instructions for:**

Ground 1 teaches [23pre] and [30pre], e.g., with a video encoding/decoding "apparatus" comprising "a processor" and "computer-readable medium, carrying instructions, which when executed[,]… causes [the processor] to carry out any of the methods disclosed herein." ASUS-1004, ¶1, ¶¶24-27, ¶114, ¶116, claims 17-19; ASUS-1003, ¶¶CB207-209, ¶¶CB230-233.

[30pre] recites "an encoder" but should recite "a decoder" consistent with the rest of the claim. ASUS-1003, ¶¶CB231-232.

19. Claims 24-28

Beyond their preambles, claims 24-28 are identical to claims 10-14. Grounds 1 and 2 teach the preambles (apparatus of claim 23) (*supra* §VI.A.18) and the rest of claims 24-28 as explained for claims 10-14, respectively. *Supra* §§VI.A.11-15; ASUS-1003, ¶¶CB223-229.

VII. PAYMENT OF FEES – 37 C.F.R. § 42.103

ASUS authorizes the Patent and Trademark Office to charge Deposit Account No. 06-1050 for the fee set in 37 C.F.R. § 42.15(a) for this Petition and further authorizes payment for any additional fees to be charged to this Deposit Account.

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

A. Real Party In Interest

The real parties-in-interest here are ASUSTeK Computer Inc. and ASUS Computer International. No other parties directed, controlled, or funded this *Inter Partes* Review proceeding (IPR).

B. Related Matters

Petitioner is not aware of any disclaimers or reexamination certificates for the ’714 patent. The ’714 patent was the subject of *inter partes* reviews in IPR2024-00604 and IPR2024-00605, both of which were terminated due to settlement. The ’714 patent is the subject of a number of civil actions including:

- *Certain Video-Capable Laptop, Desktop Computers, Handheld Computers, Tablets, Televisions, Projectors, and Components and Modules Thereof*; Inv. No. 337-TA-1448 (Violation), 337-TA-1448 (ITC), filed April 11, 2025;
- *Nokia Technologies Oy* v. *ASUSTeK Computer Inc. et al.*, 2-25-cv-03053 (CDCA), filed April 07, 2025;
- *Nokia Technologies Oy* v. *Hisense Co. Ltd. et al.*, 1-25-cv-01871 (NDGA), filed April 07, 2025;
- *Nokia Technologies Oy* v. *Acer Inc. et al.*, 1-25-cv-00523 (WDTX), filed April 07, 2025;
- *Element Television Company, LLC et al* v. *Nokia Corporation a/k/a Nokia of America Corporation et al.*, 0-24-cv-04269 (DMN), filed November 25, 2024;
- *Amazon.com, Inc. et al* v. *Nokia Technologies Oy*, IPR2024-00604 (PTAB), filed April 08, 2024;
- *Amazon.com, Inc. et al* v. *Nokia Technologies Oy*, IPR2024-00605 (PTAB), filed April 08, 2024;
- N*okia Technologies Oy* v. *HP Inc. f/k/a Hewlett-Packard Company*, 1-23-cv-01237 (DDE), filed October 31, 2023;

- *Certain Video Capable Electronic Devices, Including Computers, Streaming Devices, Televisions, and Components and Modules Thereof*; Inv. No. 337-TA-1380 (Violation), 337-TA-1380 (ITC), filed October 31, 2023;
- *Nokia Corporation et al* v. *Amazon.com, Inc.*, 1-23-cv-01232 (DDE), filed October 27, 2023.

C. Lead And Back-Up Counsel Under 37 C.F.R. § 42.8(b)(3)

ASUS provides the following designation of counsel.

Lead Counsel	Backup counsel
Jeremy J. Monaldo, Reg. No. 58,680 Fish & Richardson P.C. 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945 Email: IPR54587-0016IP1@fr.com	Kim Leung, Reg. No. 64,399 Joseph V. Colaianni, Reg. No. 39,948 Linhong Zhang, Reg. No. 64,749 Jack R. Wilson IV, Reg. No. 75,011 Fish & Richardson P.C. 60 South Sixth Street, Suite 3200 Minneapolis, MN 55402 Tel: 202-783-5070 Fax: 877-769-7945

D. Service Information

Please address all correspondence and service to the address listed above.

Petitioner consents to electronic service by email at IPR54587-0016IP1@fr.com.

Respectfully submitted,

Dated: June 16, 2025

/Jeremy J. Monaldo/
Jeremy Monaldo, Reg. No. 58,680
Kim Leung, Reg. No. 64,399
Joseph V. Colaianni, Reg. No. 39,948
Linhong Zhang, Reg. No. 64,749
Jack R. Wilson IV, Reg. No. 75,011
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945
IPR54587-0016IP1@fr.com

(Control No. IPR2025-01153)

Attorneys for Petitioner

CERTIFICATION UNDER 37 CFR § 42.24

Under the provisions of 37 CFR § 42.24(d), the undersigned hereby certifies that the word count for the foregoing Petition for *Inter Partes* Review totals 13,964 words, which is less than the 14,000 allowed under 37 CFR § 42.24.

Dated: June 16, 2025

/Jeremy J. Monaldo/

Jeremy Monaldo, Reg. No. 58,680
Kim Leung, Reg. No. 64,399
Joseph V. Colaianni, Reg. No. 39,948
Linhong Zhang, Reg. No. 64,749
Jack R. Wilson IV, Reg. No. 75,011
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
T: 202-783-5070
F: 877-769-7945
IPR54587-0016IP1@fr.com

Attorneys for Petitioner

CERTIFICATE OF SERVICE

Pursuant to 37 CFR §§ 42.6(e)(4)(i) *et seq.* and 42.105(b), the undersigned certifies that on June 16, 2025, a complete and entire copy of this Petition for *Inter partes* Review, Power of Attorney, and all supporting exhibits were provided via Federal Express, to the Patent Owner by serving the correspondence address of record as follows:

Nokia Corporation and Alston & Bird LLP
Vantage South End
1120 South Tryon Street
Suite 300
Charlotte, NC 28203-6818

/Hoi Cheung/
Hoi Cheung
Fish & Richardson P.C.
60 South Sixth Street, Suite 3200
Minneapolis, MN 55402
hcheung@fr.com