EXHIBIT D-4

RUSERT ALONE OR IN COMBINATION WITH KARCZEWICZ AND/OR LIN

ANTICIPATES / RENDERS OBVIOUS U.S. PATENT NO. 10,536,714

As described in the following claim chart, claims 1-30 of U.S. Patent No. 10,536,714 (the "'714 patent") are invalid in view of US Patent Application Publication 2011/0194609 ("Rusert"), entitled "Selecting Predicted Motion Vector Candidates", published on August 11, 2011. Rusert was published on August 11, 2011, and is prior art under at least 35 U.S.C. § 102(a) and 35 U.S.C. § 102(e).

To the extent that these Invalidity Contentions rely on or otherwise embody particular constructions of terms or phrases in the Asserted Claims, Respondents are not proposing any such constructions as proper constructions of those terms or phrases. Various positions put forth in this document are predicated on Complainants' incorrect and overly broad interpretation of its claims. Those positions are not intended to and do not necessarily reflect Respondents' interpretation of the true and proper scope of Complainants' claims, and Respondents reserve the right to adopt claim construction positions that differ from or even conflict with various positions put forth in this document.

To the extent the Rusert does not anticipate the claims of the '714 patent, the asserted claims of the '714 patent would have been obvious in view of Rusert in combination with U.S. Patent Application Publication No. 2011/0249721 to Karczewicz et al ("Karczewicz"). Karczewicz was filed April 11, 2011, published October 2011, and is prior art under §§102(a) and (e). Karczewicz teaches aspects of H.265 drafts including terminology for coding and prediction units (PUs). Ex-1005, ¶¶32-35. Karczewicz is analogous art in the same field and teaches techniques for "coding video data." *E.g.*, Karczewicz, Abstract.

As demonstrated herein, 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.

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." Ex-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. Karczewicz, ¶32, ¶¶35-37, ¶66, ¶71. Karczewicz teaches H.265 concepts including PUs. Therefore, this combination provides express motivation to apply Rusert's teachings to other standards including H.265, as taught by Karczewicz.

Nokia Exhibit 2006, p. 1
ASUS v. Nokia
IPR2025-01153

The combination of Rusert and Karczewicz 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. Rusert, ¶¶2-5, ¶36, ¶43. As Karczewicz explains, for H.265 drafts, motion vectors and motion prediction operated on blocks called PUs. Karczewicz, ¶¶33-36, ¶¶64-66. Karczewicz also teaches types of information conveyed by motion vectors. Karczewicz, ¶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. Rusert, ¶7; Karczewicz, ¶6.

Additionally, a POSITA would have found motivation in the similarity of the references. Both are directed to ITU video-coding standards. Rusert, ¶116; Karczewicz, ¶¶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. Karczewicz, ¶33, ¶35.

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. Rusert, ¶¶3-4, ¶67, ¶77, ¶116.

The combination of Rusert and Karczewicz 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. Rusert, ¶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).

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. Karczewicz, ¶5. ¶38. H.265 does not prevent H.264 teachings from being applied, or vice-versa.

The combination of Rusert and Karczewicz would have had predictable results. Rusert already applies its teachings to block-based encoding/decoding. Rusert, ¶2, ¶11. A PU is a type of block. Karczewicz, ¶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). Rusert, ¶116. 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. Rusert, ¶116.

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. Rusert, ¶116. Karczewicz complements Rusert by teaching H.265 terminology and concepts. Karczewicz, ¶32. The combination of Rusert and Karczewicz 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. Rusert, ¶11; Karczewicz, ¶2.

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.

The asserted claims of the '714 patent are additionally rendered obvious in view of Rusert, Karczewicz, and U.S. Patent Application Publication No. 2014/0092981 to Lin et al ("Lin"). Lin claims priority to provisional 61/500,903 ("Lin Provisional"), 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 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:	Lin Provisional, 7, 9, 12
determining neighboring blocks of the current block, wherein an MVP candidate set is derived from MVP candidates associated with the neighboring blocks;	*Id.* at 16 (Fig. 6)
determining at least one redundant MVP candidate according to a non-MV-value based criterion;	*Id.* at 9-11
removing said at least one redundant MVP candidate from the MVP candidate set; and	*Id.* at 9-11
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.	*Id.* at 8-11

Nokia Exhibit 2006, p. 3
ASUS v. Nokia
IPR2025-01153

Lin teaches removal of redundant MVPs for divided coding units (CUs), which is relevant to claim 4, among other dependent claims. Lin, ¶25. Lin is analogous art in the same field and teaches techniques "for motion vector coding." Lin, ¶2. A PHOSITA would have been motivated to combine Lin with Rusert and Karczewicz because a CU comprises one prediction unit (PU) when it has uniform motion and is only divided if different parts are moving in different directions, as Karczewicz explains. See Lin, ¶4; Lin Provisional, 000007; Karczewicz, ¶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. Lin, ¶25, ¶44; Lin Provisional, 000010. This would have had the predictable result of excluding such candidates, consistent with the reason why a block was divided to begin with.

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[.]" Rusert, ¶12, ¶7. Therefore, the combination of Rusert, Karczewicz, and Lin 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. Rusert, ¶12. In addition, the combination would not have changed the principle of operation for any reference, in the manner taught by each reference. Lin, ¶44; Lin provisional, 000010.

Further, a POSITA would have had a reasonable expectation of success in combining Rusert, Karczewicz, and Lin because all three references teach aspects of block-based video encoding for H.264/H.265 to reduce PMV candidates. Rusert, ¶25, ¶39, ¶¶42-44; Karczewicz, ¶3, ¶5, ¶38; Lin, ¶4, ¶25; Lin Provisional, 000009-11. The combination does not modify Rusert, Karczewicz, or Lin in a way that would render any reference inoperative. In addition, the underlying rationale for Lin's teachings is broadly applicable to scenarios when a block has been divided because blocks are divided into multiple PUs to assign them different motion vectors. Karczewicz, ¶35; Lin, ¶4; Lin Provisional, 000007.

Nokia Exhibit 2006, p. 4
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
1[preamble] A method comprising:	Rusert, as evidenced by the example citations below, discloses a method. *See, e.g.,* Rusert, Abstract: > There is provided a method of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from a current block. The method comprises identifying allowed distance values of distances between the current block and the previous block. The method further comprises selecting a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0001: > The present application relates to a method of selecting PMV candidates, a video encoding apparatus, a video decoding apparatus, and a computer-readable medium. Rusert, ¶0011 > Accordingly, there is provided a method of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from a current block. The method comprises identifying allowed distance values of distances between the current block and the previous block. The method further comprises selecting a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block.

Nokia Exhibit 2006, p. 5
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.
1[a] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* Rusert, ¶0003: > At high compression ratios (or equivalently, low video bitrates), motion vector coding takes a large part of the total amount of bits, especially in recent video coding standards such as H.264/AVC where small motion compensation block sizes are used. Typically, lossless predictive coding of motion vectors is used, i.e. coding of a motion vector MV consists of first building a motion vector predictor PMV for the vector to be coded and then transmitting the difference: DMV, where DMV=MV−PMV between the motion vector and the motion vector predictor.

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	Rusert, ¶0011: Accordingly, there is provided a method of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from a current block. The method comprises identifying allowed distance values of distances between the current block and the previous block. The method further comprises selecting a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0012: A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. Rusert, ¶0015: The method may further comprise identifying allowed positions of previous blocks, and wherein the selected set of PMV candidates comprises a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block and an allowed position. Rusert, ¶0024: There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0025: There is further provided a video decoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0034: FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. Rusert, ¶0036-49: [0036] The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV.

Nokia Exhibit 2006, p. 8
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). [0037] The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. [0038] There are two major operations used for construction of the set of PMV candidates 240, initialization and update. [0039] Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive the list PMV_CANDS in the same way as the encoder. [0040] Alternatively, one or more motion vector candidates that have not been encoded previously may be added into the PMV_CANDS list at the encoder, and then those motion vectors will be explicitly signaled to the decoder for use with PMV_CANDS, such that PMV_CANDS can be updated in the same way both at the encoder and the decoder.

Nokia Exhibit 2006, p. 9
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0041] The PMV_CANDS list used for coding a motion vector associated with a current motion compensation block can be dynamically generated specifically for the current motion compensation block, i.e. without consideration of the PMV_CANDS lists used for coding of MVs associated with motion compensation blocks previously coded. In that case, before a block is processed, a PMV_CANDS list is initialized and then updated with a number of previously coded or pre-defined motion vectors. Alternatively, a PMV_CANDS list may be initialized once (for example before the start of video encoding/decoder, or before a frame is processed or after a number of macroblocks have been encoded in a frame), and then used for coding of more than one motion vector, the advantage being that the possibly complex process of deriving the PMV_CANDS list need only be processed once for coding of a set of motion vectors. When being used for coding of more than one motion vector, the PMV_CANDS list may however be updated after coding one of the motion vectors. For example, the PMV_CANDS list may first be used for coding of a motion vector MV associated with a first motion compensation block, then PMV_CANDS may be updated using the vector MV (e.g. MV is added into the list), and then used for coding of a second motion compensation block. By subsequently updating PMV_CANDS with coded motion vectors, the list is updated according to a sliding window approach. [0042] During encoding of a video, one or multiple PMV_CANDS lists may be maintained according to the sliding window approach, e.g. one for each frame type, one for each macroblock type, or one for each reference frame. When coding the two motion vectors associated with a bi-predicted motion compensation block, either a single or two different PMV_SAND may be used [0043] Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks. [0044] The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: [0045] all blocks of the current frame have been scanned, [0046] all blocks of a pre-defined number of subsequent frames (e.g. the last frame) have been scanned [0047] once a certain distance has been reached, [0048] as soon as a pre-defined number of unique PMV candidates have been found, [0049] all blocks of a predetermined scan pattern have been scanned. Rusert, ¶0113, Fig. 6:

Nokia Exhibit 2006, p. 11
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

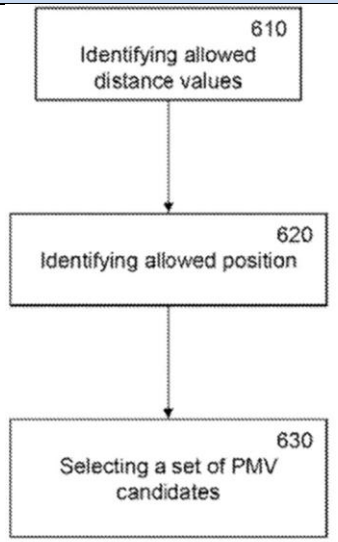


Fig. 6

> FIG. 6 illustrates a method according to the present application. At 610 allowed distance values are identified for the distances between a current block and a previous block. The previous block may have a motion vector associated therewith which was used for coding said previous block. Optionally, at 620 allowed positions for previous blocks may be identified. Such allowed positions may comprise middle and corner positions in a square of blocks centered on the current block. At 630 a set of PMV candidates is selected as a subset of the previously used motion vectors associated with previous blocks having allowed distance values, and, if optional step 620 is implemented, allowed positions. The set of PMV candidates may then used for coding of the current block.

Rusert, ¶0006:

> The number of candidates in PMV_SANDS, i.e. the "length" of PMV_SANDS, has a major impact on coding efficiency, too. The reason is that the higher the number of candidates, the

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	higher the number of bits required for signaling one of the candidates, which in turn causes additional overhead and thus reduced compression efficiency. Existing approaches assume a fixed number of candidates in PMV_SANDS (e.g. the spatially neighboring motion vectors) valid for coding of a video frame or sequence, and the number of candidates may only be reduced if some of the candidates are identical. Rusert, ¶0004: In H.264/AVG, a PMV is derived as the median of the motion vectors of three spatially neighboring blocks. Other approaches consider also temporally neighboring blocks (i.e. co-located in neighboring frames) for motion vector prediction. Instead of using a fixed rule for building PMV, recently approaches have been presented that explicitly signal a PMV to be used out of a set of PMV candidates, PMV_CANDS. Although this requires additional bits to signal one candidate out of the set, it can overall save bits for motion vector coding, since DMV coding can be more efficient. That is, identifying a PMV and signaling DMV can take fewer bits than independently signaling the MV. Rusert, ¶0005: The efficiency of motion vector coding schemes which use PMV candidate signaling depends on the suitability of the available candidates in PMV_CANDS. That is, the construction of the candidate list has a major impact on the coding performance. Existing approaches typically use motion vectors from spatially surrounding blocks or temporally neighboring blocks (co- located blocks in neighboring frames). Such construction of PMV_CANDS, i.e. considering only the few surrounding blocks as a source of motion vector predictors, can be sub-optimal. Rusert, ¶0067:

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	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, e.g. a H.264/AVC-style median predictor derived based on spatially neighboring blocks. Rusert, ¶0059, Fig. 3g: However, this ordering does have some problems. For instance, although blocks 3 and 4 have the same Chebyshev distance, block 4 has a lower Euclidian distance and so it would be appropriate to exchange the order of these two blocks. This problem is solved by using an order which is first determined by Chebyshev distance, and second by Euclidian distance. To clarify if two objects have different Chebyshev distance, the item with the smaller Chebyshev distance will always end up before the other one in the order. However, for items of the same Chebyshev distance, the order is determined by Euclidian distance. If two items have the same Chebyshev distance and the same Euclidian distance, a tie-breaking rule will be used. This provides an accurate ordering of the blocks that is still somewhat simple to calculate. An example of this ordering scheme will now be described. FIG. 3 f shows the Chebyshev distances of blocks up to three rows or columns from a central current block. As shown in FIG. 3 f, this gives several "layers" or squares of blocks with common Chebyshev distance. Further ordering can be provided by using a for-loop going over each layer (Chebyshev distance=1, 2, 3, etc.). In fact, since the blocks to the right and below the current block position have not been visited yet, and do not yet have motion vectors known for them, we can remove them from consideration. The resulting pattern is shown in FIG. 3 g. 3333333 3222223 3211123 321.--- 321---- 322---- 333---- Fig. 3g Rusert, ¶0044:

Nokia Exhibit 2006, p. 14
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶0050-¶0066: [0050] Note that sorting the list on distances may be avoided in an outwards going scan, by inserting unique vectors at the end of the PMV_CANDS list, the list is kept sorted with the spatially closest vector first in the list. [0051] In one embodiment the ordering is based on Euclidian distance to the block in question. Assume for instance that the block we wanted to encode was in position x=3 y=5. Then we would rank the surrounding blocks according to euclidian_dist=sqrt((xpos−3)^2+(ypos−5)^2). [0052] For instance, a block in position (2,1) would have a distance sqrt(1^2+4^2) sqrt(17). Since the square-root does not change the order it can be excluded. Thus, if the block marked with a dot in FIG. 3 a is the one to be coded, the surrounding blocks would have the (squared) distances shown in FIG. 3 a (this diagram shows only the nearest neighbors and, in practice would be extended outwards to the edge of the frame.

Nokia Exhibit 2006, p. 15
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0053] We then select blocks in an order that took the closest ones first, with some tie-breaking rule for blocks at the same distance. A possible order for the blocks shown in FIG. 3 a is shown in FIG. 3 b, where the order is shown according to: 1, 2, 3, . . . , 9, a, b, ... [0054] Of course some of these would never be available, such as "3" and "b" since they are directly to the right of the block to be coded, and would therefore be coded after the present block. Other blocks would sometimes be available depending upon the traversal pattern used. Some blocks may not be available due to the simple fact that there was no motion vector present in that block, or that the motion vector present was the same as a block earlier in the sequence. After these considerations it is possible that only a subset of blocks have valid motion vectors. An example of such a subset is shown in FIG. 3 c. In this case the list would consist of the motion vectors related to the shown blocks in the following order: 1, 2, 9, h, j, l, m. [0055] A problem with this pattern is that it is quite hard to "spiral outwards", that is: it is not easy to calculate which x,y-coordinate a candidate with rank k has. Of course, they could be stored in a long list (since mapping is deterministic) but that list could be huge. [0056] Accordingly, an alternative embodiment is provided using an alternative structure for which it is easier to calculate the x,y-position of an element of a certain rank. This achieved by changing the distance metric from Euclidian distance to Chebyshev distance (also known as "box distance"). The new distance is instead calculated as Chebyshev_dist=max(|xpos−xcurrent|,|ypos−ycurrent|), [0057] In the example used above, where (xcurrent, ycurrent) (3, 5), a block in position (2,1) would have Chebyshev distance=4, since

Nokia Exhibit 2006, p. 16
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

$$\text{Chebyshev_dist} = \max(|2-3|, |1-5|)$$
$$= \max(|-1|, |-4|)$$
$$= \max(1, 4)$$
$$= 4.$$

[0058] The surrounding blocks would have the Chebyshev distances shown in FIG. 3 d. This shows what looks like a square inside another square, or a box inside a box, hence the name box distance. Using Chebyshev distance it is a simple procedure to scan the candidates in order, for instance by first taking the top row, and then continuing anti-clockwise, as shown in FIG. 3 e. This illustrates a spiral scan, because the scan order spirals outwards.

[0059] However, this ordering does have some problems. For instance, although blocks 3 and 4 have the same Chebyshev distance, block 4 has a lower Euclidian distance and so it would be appropriate to exchange the order of these two blocks. This problem is solved by using an order which is first determined by Chebyshev distance, and second by Euclidian distance. To clarify if two objects have different Chebyshev distance, the item with the smaller Chebyshev distance will always end up before the other one in the order. However, for items of the same Chebyshev distance, the order is determined by Euclidian distance. If two items have the same Chebyshev distance and the same Euclidian distance, a tie-breaking rule will be used. This provides an accurate ordering of the blocks that is still somewhat simple to calculate. An example of this ordering scheme will now be described. FIG. 3 f shows the Chebyshev distances of blocks up to three rows or columns from a central current block. As shown in FIG. 3 f, this gives several “layers” or squares of blocks with common Chebyshev distance. Further ordering can be provided by using a for-loop going over each layer (Chebyshev distance=1, 2, 3, etc.). In fact, since the blocks to the right and below the current block position have not been visited yet, and do not yet have motion vectors known for them, we can remove them from consideration. The resulting pattern is shown in FIG. 3 g.

[0060] The remaining blocks are ordered as follows. Consider the layer of blocks with Chebyshev distance of 3. We then know that the items in the middle of each side of the layer

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	will be the ones closest to the current block in a Euclidean sense, so we start by visiting them in some order, as shown in FIG. 3 h. The second closest blocks will be the ones right next to these, so we proceed to next apply an order to these as shown in FIG. 3 i. The next closest blocks will be those next to the ones just ordered as shown in FIG. 3 j. Finally, the three corner blocks are placed in the order as shown in FIG. 3 k. [0061] We have now a finished layer 3, and would need to proceed to layer 4 if we had not found the required number of different motion vectors. A complete ordering for the blocks up to Chebyshev distance 4 is shown in FIG. 3 l, where the order is shown as 1, 2, . . . 9, a, b, . . . z, A, B, . . . U. [0062] A problem with this arrangement is that sometimes motion vectors close to the current block are all exactly the same. It has been identified as advantageous to discard such duplicates. However, a problem with the scan pattern of FIG. 3 l is that a large number of the nearest neighbor backs must be investigated before any duplicates may be discarded. This means that a large number of vectors must sometimes be investigated before the candidate list can be filled and as such the above scan pattern can be considered to be slow. [0063] Another disadvantage with the scan pattern of FIG. 3 l is that motion vectors close to each other may be similar to each other even if they are not exactly the same. Thus the resulting list of PMVs may lack the diversity necessary encode motion vectors efficiently. For instance, the current block may be part of the background whereas all MVs 1, 2, . . . , 9, a, b, . . . , z, A, B, . . . , U may be part of a moving object. It could then be advantageous to be able to reach MVs corresponding to b her away (e.g., in layer 4, 5 etc) before filling up the list. [0064] Accordingly, a further embodiment is provided which can be said to speed up the scan pattern. In this embodiment entire layers of blocks having common Chebyshev distance are excluded from consideration. For instance, we could use only layers that are powers of two: 1, 2, 4, 8,.....This is an example of an exponential increase, but any series which grows greater than a linear increase will provide a good trade off between a selection of blocks close to the

Nokia Exhibit 2006, p. 18
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	current block and a few blocks a significant distance away. Using Chebyshev distances of 1, 2, 4, 8 etc, the ordering in this embodiment is shown in FIG. 3 m. [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, Figs. 3a-3n:

Nokia Exhibit 2006, p. 19
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

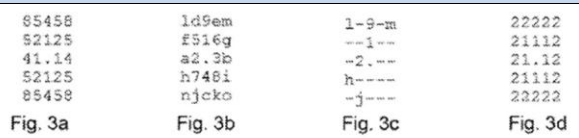


Fig. 3a Fig. 3b Fig. 3c Fig. 3d

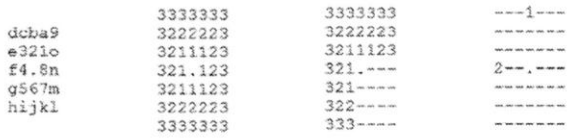


Fig. 3e Fig. 3f Fig. 3g Fig. 3h



Fig. 3i Fig. 3j Fig. 3k



Fig. 3l Fig. 3m Fig. 3n

Rusert, ¶0041:

The PMV_CANDS list used for coding a motion vector associated with a current motion

Nokia Exhibit 2006, p. 20
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	compensation block can be dynamically generated specifically for the current motion compensation block, i.e. without consideration of the PMV_CANDS lists used for coding of MVs associated with motion compensation blocks previously coded. In that case, before a block is processed, a PMV_CANDS list is initialized and then updated with a number of previously coded or pre-defined motion vectors. Alternatively, a PMV_CANDS list may be initialized once (for example before the start of video encoding/decoder, or before a frame is processed or after a number of macroblocks have been encoded in a frame), and then used for coding of more than one motion vector, the advantage being that the possibly complex process of deriving the PMV_CANDS list need only be processed once for coding of a set of motion vectors. When being used for coding of more than one motion vector, the PMV_CANDS list may however be updated after coding one of the motion vectors. For example, the PMV_CANDS list may first be used for coding of a motion vector MV associated with a first motion compensation block, then PMV_CANDS may be updated using the vector MV (e.g. MV is added into the list), and then used for coding of a second motion compensation block. By subsequently updating PMV_CANDS with coded motion vectors, the list is updated according to a sliding window approach. Rusert, ¶0042: During encoding of a video, one or multiple PMV_CANDS lists may be maintained according to the sliding window approach, e.g. one for each frame type, one for each macroblock type, or one for each reference frame. When coding the two motion vectors associated with a bi-predicted motion compensation block, either a single or two different PMV_SAND may be used Rusert, ¶0038: There are two major operations used for construction of the set of PMV candidates 240, initialization and update.

Nokia Exhibit 2006, p. 21
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0116: Further, while examples have been given in the context of particular coding standards, these examples are not intended to be the limit of the coding standards to which the disclosed method and apparatus may be applied. For example, while specific examples have been given in the context of H.264/AVC, the principles disclosed herein can also be applied to an MPEG2 system, other coding standard, and indeed any coding system which uses predicted motion vectors. Rusert, ¶0037: The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. Rusert, ¶0106: Now, if cost(A_index)−cost(B_index)+cost_x(D−1)−3<=0, we do not have to send the sign bit for candidate B. As an example, if A=(11,2), B=(13,2) and A_index is 1 and B_index is 0001, then 0=2 and cost(A_index)−cost(B_index)+cost_x(D−1)−3 equals 1−4+3−3=−2 which is smaller than 0, hence we do not need to send the sign bit for B. This example is illustrated in FIG. 5, where the x and y axis show x and y components of motion vectors. Each box in FIG. 5 represents a motion vector: the bit cost of coding the respective motion vector using PMV candidate A is shown to the left of the box and the bit cost of coding the respective motion vector using PMV candidate B is shown to the right of the box. From FIG. 5 it can be seen that for MVs with an x component of 12 or less the most efficient PMV to use is A, whereas for MVs with an x component of 13 or more, the most efficient PMV to

Nokia Exhibit 2006, p. 22
ASUS v. Nokia
IPR2025-01153

use is B.

Rusert, ¶0036:

> The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block).

Rusert, ¶¶0091-0094:

[0091] As an example, assume we have the following candidates:

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

[0092] Also, assume that we encode a difference, DMV, (xdiff, ydiff) where xdiff and ydiff are encoded using Table 3 below. If we want to encode a motion vector, such as MV=(0,2), we can then encode it using candidate 3, which is PMV=(0,2), plus a difference DMV=(0,0):

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	PMV+DMV=MV (0,2)+(0,0)=(0,2). [0093] The index costs four bits (the code length for index 3 is four bits), and each of the zeroes in the difference cost one bit, so the total number of bits required to code MV=(0,2) is 4+1+1=6 bits. [0094] However, we can also code the vector using index 0, PMV=(−1,2), plus a difference DMV=(1,0): (−1,2)+(1,0)=(0,2). Rusert, ¶0100: Assume that we want to encode a vector using PMV candidate having index number 3, PMV=(11,3). Since the PMV candidate with index number 4 is to the right of it (having coordinates PMV=(12,3)), it is advantageous to encode it with candidate 4 instead if the x-coordinate is 12 or greater. As an example, the vector MV=(15,2) can be encoded using candidate 3 as (11,3)+(4,−1) costing four bits for the index, seven bits for +4 and three bits for −1 (see table 3), in total 14 bits. But it can also be encoded using candidate 4 as (12,3)+(3,−1) which costs four bits for the index, five bits for +3 and three bits for −1, in total 12 bits. Since candidate 4 will always be closer to any point in the right half-plane, it will be advantageous to choose candidate 4 for that. Likewise, it is better to choose candidate 3 if we are in the left

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	half-plane (as seen from the point between (11,3) and (12,3)). Rusert, ¶0002: Recent video coding standards are based on the hybrid coding principle, which comprises motion compensated temporal prediction of video frames and coding of frame residual signals. For efficient motion compensated temporal prediction, block-based motion models are used to describe the motion of pixel blocks across frames. Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames). These motion vectors are coded in the video bit stream along with the frame residual signals. Rusert, ¶0035: The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶0075: How to determine the order of motion vector candidates in candidate list will now be addressed. It is assumed that the candidates in PMV_CANDS are sorted, and use of one of the candidates in PMV_CANDS for prediction is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. The following methods can be used

Nokia Exhibit 2006, p. 25
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	when updating a PMV_CANDS list in order to sort the candidates in a way that is beneficial for overall coding efficiency. Rusert, ¶0088-102, and included Tables: [0088] As mentioned above, the candidates in PMV_CANDS are sorted and the use of one of the candidates in PMV_CANDS is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. Such code words can be defined e.g. according to Variable Length Coding (VLC) tables. The VLC table used can depend on the maximum number of candidates in PMV_SANDS (the list size), as e.g. dynamically adapted according to the methods above. Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list.

Nokia Exhibit 2006, p. 26
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 1

Example VLC codes for different maximum list sizes.

Maximum list size C	Index: VLC code
	0: - (unambiguous, no signaling necessary)
2	0: 0
	1: 1
3	0: 1
	1: 00
	2: 01
4	0: 1
	1: 00
	2: 010
	3: 011
5	0: 1
	1: 00
	2: 010
	3: 0110
	4: 0111
6	0: 1
	1: 010
	2: 011
	3: 001
	4: 0000
	5: 0001
7	0: 1
	1: 010
	2: 011
	3: 0010
	4: 0011
	5: 0000
	6: 0001

Nokia Exhibit 2006, p. 27
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

[0089] For bi-predicted motion compensation blocks, two motion vectors are coded, and thus two PMV candidates can be necessary. In that case the index numbers for the two PMV candidates can be coded together to further reduce the number of bits required for index coding. Table 2 shows an example for joint index coding, considering that both motion vectors use the same PMV_CANDS list, and that it is likely that both motion vectors use the same predictor in the PMV_CANDS list. Here idx0 and idx1 denote the indexes for first and second predictor, respectively. VLC0(idx,C) denotes a VLC for an index "idx" according to Table 1 considering a maximum list size of C.

TABLE 2

VLC code for coding of two candidates indexes associated with bi-predicted block, C: maximum size of PMV_CANDS.

Case	VLC code
idx1 < idx0	VLC0 (idx0,C) 0 VLC0 (idx1,C-1)
idx1 = idx0	VLC0 (idx0,C) 1
idx1 > idx0	VLC0 (idx0,C) 0 VLC0 (idx1-1,C-1)

[0090] Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful.

Nokia Exhibit 2006, p. 28
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0091] As an example, assume we have the following candidates:

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

[0092] Also, assume that we encode a difference, DMV, (xdiff, ydiff) where xdiff and ydiff are encoded using Table 3 below. If we want to encode a motion vector, such as MV=(0,2), we can then encode it using candidate 3, which is PMV=(0,2), plus a difference DMV=(0,0):

PMV+DMV=MV

(0,2)+(0,0)=(0,2).

Nokia Exhibit 2006, p. 29
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

[0093] The index costs four bits (the code length for index 3 is four bits), and each of the zeroes in the difference cost one bit, so the total number of bits required to code MV=(0,2) is 4+1+1=6 bits.

[0094] However, we can also code the vector using index 0, PMV=(−1,2), plus a difference DMV=(1,0):

(−1,2)+(1,0)=(0,2).

[0095] The index costs one bit (the code length for index 1 is one bit), the xdiff=1 term in the difference costs three bits (see Table 3 below) and the ydiff=0 term costs one bit. Hence we get 1+3+1=5 bits in total, which is better than using index 3, which required 6 bits. It is easy to see given that the vector difference is coded using Table 3 below, that it will never be beneficial to use index 3, because using index 0 will always be one hit cheaper or better.

Hence we can eliminate the candidate vector (0,2) and we get instead:

Value	Index	Code
(−1,2)	0	1
(13, 4)	1	010
(12, 3)	2	011
(3, 4)	4	001
(−4, 1)	5	0000
(4, 8)	6	0001

[0096] Index 4, vector (3,4) now has a shorter code; it is now three bits instead of four. Hence we have gained from the elimination if the vector (3,4) is used, and never lost anything. In the above example we removed the PMV candidate with index number 3, but it should be evident for a person ski d in the art that this is just an example. For instance, it may in some cases be beneficial to remove candidates 1 and 2 as well.

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

[0097] Since the same analysis is performed both in the encoder and the decoder, the same vector is removed from the list both in the encoder and the decoder. Hence, after the removal, both the encoder and the decoder will use the same candidate list.

[0098] Sometimes it may not be possible to ensure a gain by removing a single candidate, but it may be possible if two or more candidates are eliminated simultaneously. Another possibility is that altering the order of the candidates or even adding a new candidate to the list can allow us to remove candidates that have now been rendered unnecessary, and therefore be beneficial regardless of the final motion vector to be encoded,

TABLE 3

The cost of sending the differential

Differential	Code
0	1
−1	010
1	011
−2	00100
2	00101
−3	00110
3	00111
−4	0001000
4	0001001
. . .	. . .

Nokia Exhibit 2006, p. 31
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

[0099] The number of bits needed can be further reduced by not sending the sign bit, this is possible because in some circumstances the sign bit for the differential is unnecessary. Assume for example that we have the following list of PMV candidates:

Value	Index	Code
(−1, 2)	0	1
(13, 8)	1	010
(3, 4)	2	011
(11, 3)	3	0010
(12, 3)	4	0011
(1, 2)	5	0000
(4, 8)	6	0001

[0100] Assume that we want to encode a vector using PMV candidate having index number 3, PMV=(11,3). Since the PMV candidate with index number 4 is to the right of it (having coordinates PMV=(12,3)), it is advantageous to encode it with candidate 4 instead if the x-coordinate is 12 or greater. As an example, the vector MV=(15,2) can be encoded using candidate 3 as

(11,3)+(4,−1)

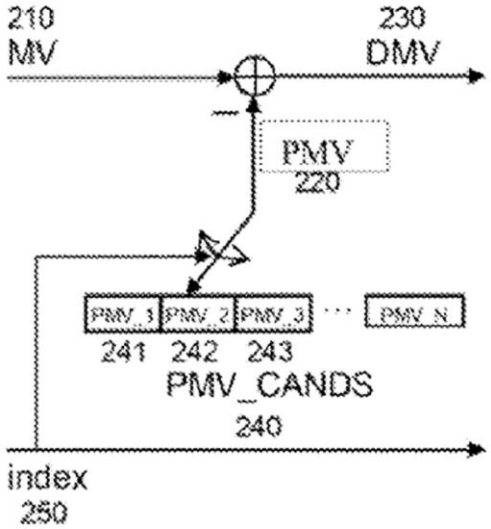
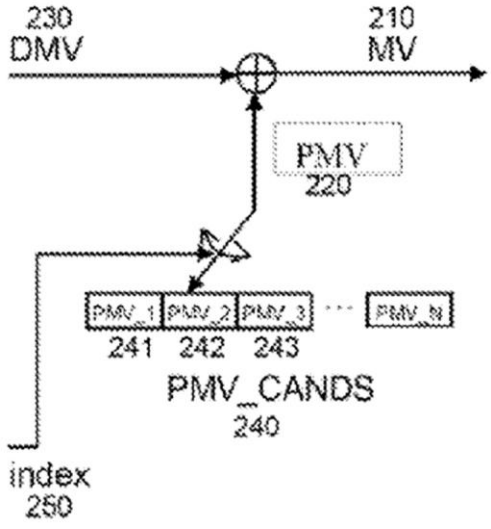
costing four bits for the index, seven bits for +4 and three bits for −1 (see table 3), in total 14 bits. But it can also be encoded using candidate 4 as

(12,3)+(3,−1)

which costs four bits for the index, five bits for +3 and three bits for −1, in total 12 bits. Since candidate 4 will always be closer to any point in the right half-plane, it will be advantageous to choose candidate 4 for that. Likewise, it is better to choose candidate 3 if we are in the left half-plane (as seen from the point between (11,3) and (12,3)).

[0101] This means that it is unnecessary to specify the sign bit for the differential in the x-component, since it will always be negative for (11.3) and positive for (12,3). The sign bit is

Nokia Exhibit 2006, p. 32
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the last bit in Table 3, except for 0 which does not have a sign bit. This means that if either candidate 3 or 4 will be selected, they will be one bit cheaper to encode. [0102] The decoder will of course do the same analysis and avoid reading the sign bit if the above situation has occurred. Rusert, Fig. 2a-2b:  Fig. 2a  Fig. 2b See also the disclosures cited for limitations 1[c] and 1[d], *infra.* The disclosures of Karczewicz are also relevant to this limitation.

Nokia Exhibit 2006, p. 33
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	See, e.g., Karczewicz, ¶¶ 33-36: [0033] HM refers to a block of video data as a coding unit (CU). Syntax data within a bitstream may define a largest coding unit (LCU), which is a largest coding unit in terms of the number of pixels. In general, a CU has a similar purpose to a macroblock of H.264, except that a CU does not have the same size limitations associated with macroblocks. For example, a CU may be split into sub-CUs, and the size of the LCU may be different depending on the scenario or situation. In general, references in this disclosure to a CU may refer to a LCU of a picture or a sub-CU of an LCU. A LCU may be split into sub-CUs, and each sub-CU may be split into sub-CUs. Syntax data for a bitstream may define the size of the LCU as well as a maximum number of times an LCU may be split, referred to as maximum CU depth. Accordingly, a bitstream may also define a smallest coding unit (SCU). This disclosure also uses the term "block" to refer to any of an LCU, SCU, CU, prediction unit (PU), or transform unit (TU) (PU and TU are described in further detail below). [0034] A LCU may be associated with a quadtree data structure. In general, a quadtree data structure includes one node per CU, where a root node corresponds to the LCU. If a CU is split into four sub-CUs, the node corresponding to the CU includes four child nodes, each of which corresponds to one of the sub-CUs or one or more sub-CUs of a sub-CU. Each node of the quadtree data structure may provide syntax data for the corresponding CU. For example, a node in the quadtree may include a split flag, indicating whether the CU corresponding to the node is split into sub-CUs. Syntax elements for a CU may be defined recursively, and may depend on whether the CU is split into sub-CUs. Additional syntax elements may also be included for each CU, such as flags indicating whether filtering should be applied to the CU. [0035] A CU that is not split may include one or more prediction units (PUs). In general, a PU represents all or a portion of the corresponding CU, and includes data for retrieving a reference sample for the PU. For example, when the PU is intra-mode encoded, the PU may include data describing an intra-prediction mode for the PU. As another example, when the PU is inter-mode encoded, the PU may include data defining a motion vector for the PU. The data defining the motion vector may describe, for example, a horizontal component of the motion vector, a vertical component of the motion vector, a resolution for the motion vector (e.g., one-quarter pixel precision or one-eighth pixel precision), a reference frame to

Nokia Exhibit 2006, p. 34
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	which the motion vector points, and/or a reference list (e.g., list 0 or list 1) for the motion vector. Data for the CU defining the PU(s) may also describe, for example, partitioning of the CU into one or more PUs. Partitioning modes may differ between whether the CU is intra- prediction mode encoded, or inter-prediction mode encoded. [0036] A CU may also include one or more transform units (TUs) that may be used to perform transform operations on video data. Following prediction using a PU, a video encoder may calculate residual values for the portion of the CU corresponding to the PU. The residual values may be transformed, scanned, and quantized. A TU is not necessarily limited to the size of a PU. Thus, TUs may be larger or smaller than corresponding PUs for the same CU. In some examples, the maximum size of a TU may correspond to the size of the corresponding CU. Karczewicz, ¶ 50: In this disclosure, the term "coding" refers to encoding or decoding. Similarly, the term "coder" generally refers to any video encoder, video decoder, or combined encoder/decoder (codec). Accordingly, the term "coder" is used herein to refer to a specialized computer device or apparatus that performs video encoding or video decoding. The CBP coding techniques of this disclosure may be applicable to encoders or decoders. The encoder uses the described VLC techniques to encode the CBP, and the decoder uses reciprocal VLC techniques to decode the CBP. Karczewicz, ¶¶ 64, 66: [0064] In this disclosure, the phrase "block" refers to any size or type of video block. For example, the phrase "block" may refer to one or more of a macroblock, LCU, CU, sub-CU, TU, or PU. Moreover, blocks may refer to blocks of video data in the pixel domain, or blocks of data in a transform domain such as a discrete cosine transform (DCT) domain, a domain similar to DCT, a wavelet domain, or the like. In addition, the phrase block may refer to the luma and chroma blocks that are generally residual blocks of data in the transform domain.

Nokia Exhibit 2006, p. 35
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0066] After generating the predictive block, the differences between the current video block being coded and the predictive block are coded as a residual block, and prediction syntax (such as a motion vector) is used to identify the predictive block. The residual block may be transformed and quantized. Transform techniques may comprise a DCT process or conceptually similar process, integer transforms, wavelet transforms, or other types of transforms. In a DCT or DCT-like process, as an example, the transform process converts a set of pixel values (e.g., residual values) into transform coefficients, which may represent the energy of the pixel values in the frequency domain. Quantization is typically applied on the transform coefficients, and generally involves a process that limits the number of bits associated with any given transform coefficient. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
1[b] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* Rusert, ¶0011: Accordingly, there is provided a method of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from a current block. The method comprises identifying allowed distance values of distances between the current block and the previous block. The method further comprises selecting a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block.

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0012: A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. Rusert, ¶0024: There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0025: There is further provided a video decoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block.

Nokia Exhibit 2006, p. 37
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0013-17: [0013] Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. [0014] At least one group of PMV candidates may have a common distance value. The method disclosed herein may also be used for managing PMV candidates. [0015] The method may further comprise identifying allowed positions of previous blocks, and wherein the selected set of PMV candidates comprises a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block and an allowed position. [0016] The allowed positions may be are corner and middle block positions. A previous block at a corner position is offset from a current block position by an equal distance in a horizontal direction and a vertical direction. A previous block at a middle position is either horizontally aligned or vertically aligned with a current block position. In a system where coding is performed by horizontal lines of pixels starting in the top left corner, the allowed block positions may comprise: blocks horizontally aligned with and offset to the left of the current block; blocks vertically aligned with and offset above the current block; and blocks offset to the left and above the current block by the same distance. The allowed distance values and/or the allowed positions may be predetermined. [0017] The allowed distance values may comprise elements of a series, the elements in the series increasing at a rate greater than a linear increase. This allows the PMV candidate list to be created comprising a large number of PMV candidates associated with previous blocks near the current block, but allowing for a few PMV candidates to be included which are

Nokia Exhibit 2006, p. 38
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	associated with previous blocks distant from the current block. This is advantageous because while it is more likely that a motion vector associated with a previous block close to the current block will be the best PMV, in some circumstances a more distant previous block may provide an optimal motion vector to use as the PMV. Rusert, ¶0037: The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. Rusert, Figs. 2a and 2b:

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

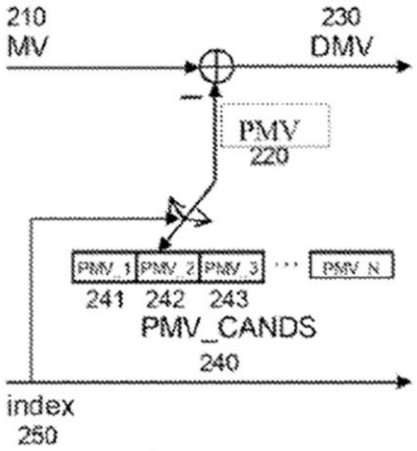


Fig. 2a

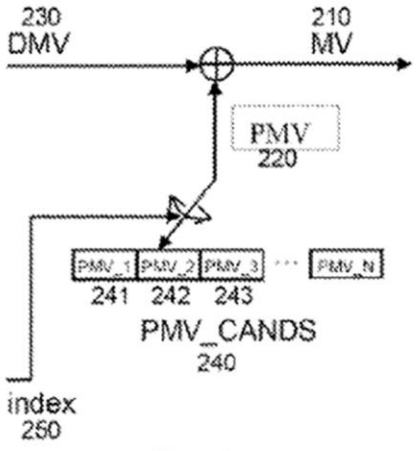


Fig. 2b

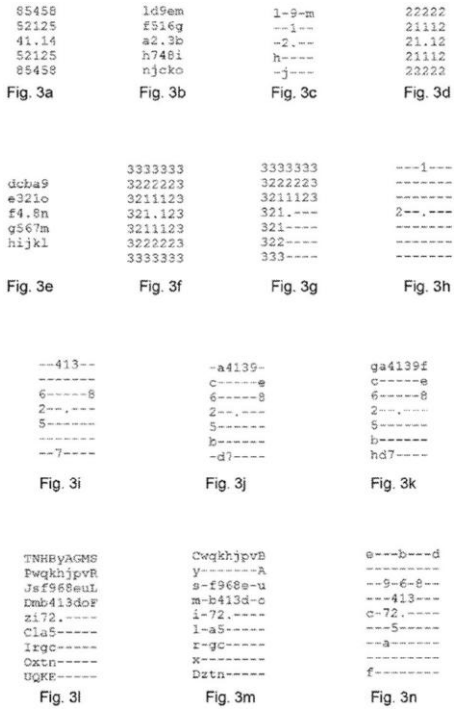
Rusert, ¶0039:

> Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive the list PMV_CANDS in the same way as the encoder.

Rusert, ¶0043:

> Before a motion vector associated with a current motion compensation block is processed, the

Nokia Exhibit 2006, p. 40
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks. Rusert, ¶0051-¶0066, Figs. 3a-3n:  Fig. 3a Fig. 3b Fig. 3c Fig. 3d Fig. 3e Fig. 3f Fig. 3g Fig. 3h Fig. 3i Fig. 3j Fig. 3k Fig. 3l Fig. 3m Fig. 3n [0051] In one embodiment the ordering is based on Euclidian distance to the block in question. Assume for instance that the block we wanted to encode was in position x=3 y=5. Then we would rank the surrounding blocks according to

Nokia Exhibit 2006, p. 41
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	euclidian_dist=sqrt((xpos−3)^2+(ypos−5)^2). [0052] For instance, a block in position (2,1) would have a distance sqrt(1^2+4^2) sqrt(17). Since the square-root does not change the order it can be excluded. Thus, if the block marked with a dot in FIG. 3 a is the one to be coded, the surrounding blocks would have the (squared) distances shown in FIG. 3 a (this diagram shows only the nearest neighbors and, in practice would be extended outwards to the edge of the frame. [0053] We then select blocks in an order that took the closest ones first, with some tie-breaking rule for blocks at the same distance. A possible order for the blocks shown in FIG. 3 a is shown in FIG. 3 b, where the order is shown according to: 1, 2, 3, . . . , 9, a, b, ... [0054] Of course some of these would never be available, such as "3" and "b" since they are directly to the right of the block to be coded, and would therefore be coded after the present block. Other blocks would sometimes be available depending upon the traversal pattern used. Some blocks may not be available due to the simple fact that there was no motion vector present in that block, or that the motion vector present was the same as a block earlier in the sequence. After these considerations it is possible that only a subset of blocks have valid motion vectors. An example of such a subset is shown in FIG. 3 c. In this case the list would consist of the motion vectors related to the shown blocks in the following order: 1, 2, 9, h, j, l, m. [0055] A problem with this pattern is that it is quite hard to "spiral outwards", that is: it is not easy to calculate which x,y-coordinate a candidate with rank k has. Of course, they could be stored in a long list (since mapping is deterministic) but that list could be huge. [0056] Accordingly, an alternative embodiment is provided using an alternative structure for which it is easier to calculate the x,y-position of an element of a certain rank. This achieved by changing the distance metric from Euclidian distance to Chebyshev distance (also known as "box distance"). The new distance is instead calculated as *Chebyshev_dist=max(|xpos−xcurrent|,|ypos−ycurrent|),*

Nokia Exhibit 2006, p. 42
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0057] In the example used above, where (xcurrent, ycurrent) (3, 5), a block in position (2,1) would have Chebyshev distance=4, since $\text{Chebyshev_dist} = \max(\lvert 2-3\rvert, \lvert 1-5\rvert)$ $= \max(\lvert -1\rvert, \lvert -4\rvert)$ $= \max(1, 4)$ $= 4.$ [0058] The surrounding blocks would have the Chebyshev distances shown in FIG. 3 d. This shows what looks like a square inside another square, or a box inside a box, hence the name box distance. Using Chebyshev distance it is a simple procedure to scan the candidates in order, for instance by first taking the top row, and then continuing anti-clockwise, as shown in FIG. 3 e. This illustrates a spiral scan, because the scan order spirals outwards. [0059] However, this ordering does have some problems. For instance, although blocks 3 and 4 have the same Chebyshev distance, block 4 has a lower Euclidian distance and so it would be appropriate to exchange the order of these two blocks. This problem is solved by using an order which is first determined by Chebyshev distance, and second by Euclidian distance. To clarify if two objects have different Chebyshev distance, the item with the smaller Chebyshev distance will always end up before the other one in the order. However, for items of the same Chebyshev distance, the order is determined by Euclidian distance. If two items have the same Chebyshev distance and the same Euclidian distance, a tie-breaking rule will be used. This provides an accurate ordering of the blocks that is still somewhat simple to calculate. An example of this ordering scheme will now be described. FIG. 3 f shows the Chebyshev distances of blocks up to three rows or columns from a central current block. As shown in FIG. 3 f, this gives several “layers” or squares of blocks with common Chebyshev distance. Further ordering can be provided by using a for-loop going over each layer (Chebyshev distance=1, 2, 3, etc.). In fact, since the blocks to the right and below the current block

Nokia Exhibit 2006, p. 43
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	position have not been visited yet, and do not yet have motion vectors known for them, we can remove them from consideration. The resulting pattern is shown in FIG. 3 g. [0060] The remaining blocks are ordered as follows. Consider the layer of blocks with Chebyshev distance of 3. We then know that the items in the middle of each side of the layer will be the ones closest to the current block in a Euclidean sense, so we start by visiting them in some order, as shown in FIG. 3 h. The second closest blocks will be the ones right next to these, so we proceed to next apply an order to these as shown in FIG. 3 i. The next closest blocks will be those next to the ones just ordered as shown in FIG. 3 j. Finally, the three corner blocks are placed in the order as shown in FIG. 3 k. [0061] We have now a finished layer 3, and would need to proceed to layer 4 if we had not found the required number of different motion vectors. A complete ordering for the blocks up to Chebyshev distance 4 is shown in FIG. 3 l, where the order is shown as 1, 2, . . . 9, a, b, . . . z, A, B, . . . U. [0062] A problem with this arrangement is that sometimes motion vectors close to the current block are all exactly the same. It has been identified as advantageous to discard such duplicates. However, a problem with the scan pattern of FIG. 3 l is that a large number of the nearest neighbor backs must be investigated before any duplicates may be discarded. This means that a large number of vectors must sometimes be investigated before the candidate list can be filled and as such the above scan pattern can be considered to be slow. [0063] Another disadvantage with the scan pattern of FIG. 3 l is that motion vectors close to each other may be similar to each other even if they are not exactly the same. Thus the resulting list of PMVs may lack the diversity necessary encode motion vectors efficiently. For instance, the current block may be part of the background whereas all MVs 1, 2, . . . , 9, a, b, . . . , z, A, B, . . . , U may be part of a moving object. It could then be advantageous to be able to reach MVs corresponding to b her away (e.g., in layer 4, 5 etc) before filling up the list. [0064] Accordingly, a further embodiment is provided which can be said to speed up the scan

Nokia Exhibit 2006, p. 44
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	pattern. In this embodiment entire layers of blocks having common Chebyshev distance are excluded from consideration. For instance, we could use only layers that are powers of two: 1, 2, 4, 8,.....This is an example of an exponential increase, but any series which grows greater than a linear increase will provide a good trade off between a selection of blocks close to the current block and a few blocks a significant distance away. Using Chebyshev distances of 1, 2, 4, 8 etc, the ordering in this embodiment is shown in FIG. 3 m. [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0004: In H.264/AVG, a PMV is derived as the median of the motion vectors of three spatially neighboring blocks. Other approaches consider also temporally neighboring blocks (i.e. co-located in neighboring frames) for motion vector prediction. Instead of using a fixed rule for building PMV, recently approaches have been presented that explicitly signal a PMV to be used out of a set of PMV candidates, PMV_CANDS. Although this requires additional bits to signal one candidate out of the set, it can overall save bits for motion vector coding, since DMV coding can be more efficient. That is, identifying a PMV and signaling DMV can take fewer bits than independently signaling the MV. Rusert, ¶0005: The efficiency of motion vector coding schemes which use PMV candidate signaling depends

Nokia Exhibit 2006, p. 45
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	on the suitability of the available candidates in PMV_CANDS. That is, the construction of the candidate list has a major impact on the coding performance. Existing approaches typically use motion vectors from spatially surrounding blocks or temporally neighboring blocks (co-located blocks in neighboring frames). Such construction of PMV_CANDS, i.e. considering only the few surrounding blocks as a source of motion vector predictors, can be sub-optimal. Rusert, ¶0036-0039: [0036] The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). [0037] The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. [0038] There are two major operations used for construction of the set of PMV candidates 240, initialization and update. [0039] Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be

Nokia Exhibit 2006, p. 46
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive the list PMV_CANDS in the same way as the encoder. Rusert, ¶0040-49: [0040] Alternatively, one or more motion vector candidates that have not been encoded previously may be added into the PMV_CANDS list at the encoder, and then those motion vectors will be explicitly signaled to the decoder for use with PMV_CANDS, such that PMV_CANDS can be updated in the same way both at the encoder and the decoder. [0041] The PMV_CANDS list used for coding a motion vector associated with a current motion compensation block can be dynamically generated specifically for the current motion compensation block, i.e. without consideration of the PMV_CANDS lists used for coding of MVs associated with motion compensation blocks previously coded. In that case, before a block is processed, a PMV_CANDS list is initialized and then updated with a number of previously coded or pre-defined motion vectors. Alternatively, a PMV_CANDS list may be initialized once (for example before the start of video encoding/decoder, or before a frame is processed or after a number of macroblocks have been encoded in a frame), and then used for coding of more than one motion vector, the advantage being that the possibly complex process of deriving the PMV_CANDS list need only be processed once for coding of a set of motion vectors. When being used for coding of more than one motion vector, the PMV_CANDS list may however be updated after coding one of the motion vectors. For example, the PMV_CANDS list may first be used for coding of a motion vector MV associated with a first motion compensation block, then PMV_CANDS may be updated using the vector MV (e.g. MV is added into the list), and then used for coding of a second motion compensation block. By subsequently updating PMV_CANDS with coded motion vectors, the list is updated according to a sliding window approach. [0042] During encoding of a video, one or multiple PMV_CANDS lists may be maintained

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	according to the sliding window approach, e.g. one for each frame type, one for each macroblock type, or one for each reference frame. When coding the two motion vectors associated with a bi-predicted motion compensation block, either a single or two different PMV_SAND may be used [0043] Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks. [0044] The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: [0045] all blocks of the current frame have been scanned, [0046] all blocks of a pre-defined number of subsequent frames (e.g. the last frame) have been scanned [0047] once a certain distance has been reached, [0048] as soon as a pre-defined number of unique PMV candidates have been found,

Nokia Exhibit 2006, p. 48
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0049] all blocks of a predetermined scan pattern have been scanned. Rusert, ¶0013: Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. Rusert, ¶0017: The allowed distance values may comprise elements of a series, the elements in the series increasing at a rate greater than a linear increase. This allows the PMV candidate list to be created comprising a large number of PMV candidates associated with previous blocks near the current block, but allowing for a few PMV candidates to be included which are associated with previous blocks distant from the current block. This is advantageous because while it is more likely that a motion vector associated with a previous block close to the current block will be the best PMV, in some circumstances a more distant previous block may provide an optimal motion vector to use as the PMV. Rusert, ¶0113, Fig. 6:

Nokia Exhibit 2006, p. 49
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

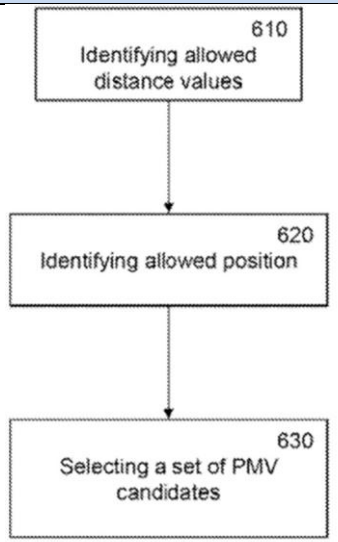


Fig. 6

FIG. 6 illustrates a method according to the present application. At 610 allowed distance values are identified for the distances between a current block and a previous block. The previous block may have a motion vector associated therewith which was used for coding said previous block. Optionally, at 620 allowed positions for previous blocks may be identified. Such allowed positions may comprise middle and corner positions in a square of blocks centered on the current block. At 630 a set of PMV candidates is selected as a subset of the previously used motion vectors associated with previous blocks having allowed distance values, and, if optional step 620 is implemented, allowed positions. The set of PMV candidates may then used for coding of the current block.

Rusert, ¶0078:

The candidates can be sorted according to frequency of occurrence of the candidate (or other

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	candidates with e.g. Euclidian or absolute distance below a pre-defined threshold) in previously coded blocks, such that vectors that describe typical motion in a video frame or sequence are assigned short code words. Alternatively, if a duplicate of a new candidate is already in the list, then the duplicate can be removed, and the new vector added at the beginning of the list, or as a further alternative the existing motion vector can be moved upwards one or more steps in the list. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. Rusert, ¶0016: The allowed positions may be are corner and middle block positions. A previous block at a corner position is offset from a current block position by an equal distance in a horizontal direction and a vertical direction. A previous block at a middle position is either horizontally aligned or vertically aligned with a current block position. In a system where coding is performed by horizontal lines of pixels starting in the top left corner, the allowed block positions may comprise: blocks horizontally aligned with and offset to the left of the current block; blocks vertically aligned with and offset above the current block; and blocks offset to the left and above the current block by the same distance. The allowed distance values and/or the allowed positions may be predetermined.

Nokia Exhibit 2006, p. 51
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0015: The method may further comprise identifying allowed positions of previous blocks, and wherein the selected set of PMV candidates comprises a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block and an allowed position. Rusert, Figs. 3a-3n:

Nokia Exhibit 2006, p. 52
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

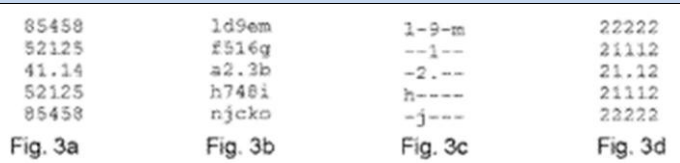


Fig. 3a Fig. 3b Fig. 3c Fig. 3d

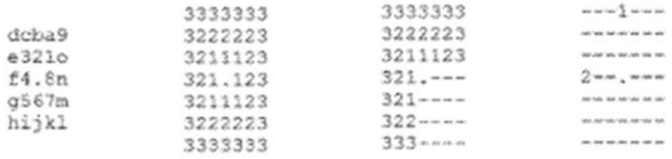


Fig. 3e Fig. 3f Fig. 3g Fig. 3h



Fig. 3i Fig. 3j Fig. 3k



Fig. 3l Fig. 3m Fig. 3n

Nokia Exhibit 2006, p. 53
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶ 0107: [0107]: In one embodiment of the proposed solution, we use a maximum of four candidates in the list. However, in another embodiment, we use seven, and there is in principle no limit to the maximum. If we allow for a bigger maximum, the list can grow and chances increase that a suitable vector can be found. On the other hand, the number of bits needed to specify the candidate vector also increases. On top of that we get the problem that many vectors can be represented using several candidates, which is unnecessary. This redundant representation grows the more vectors are added. See also the disclosures cited for limitations 1[c] and 1[d], *infra.* *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*

Nokia Exhibit 2006, p. 54
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
1[c] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* Rusert, ¶¶0002-0003: [0002] Recent video coding standards are based on the hybrid coding principle, which comprises motion compensated temporal prediction of video frames and coding of frame residual signals. For efficient motion compensated temporal prediction, block-based motion models are used to describe the motion of pixel blocks across frames. Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames). These motion vectors are coded in the video bit stream along with the frame residual signals. [0003] At high compression ratios (or equivalently, low video bitrates), motion vector coding takes a large part of the total amount of bits, especially in recent video coding standards such as H.264/AVC where small motion compensation block sizes are used. Typically, lossless predictive coding of motion vectors is used, i.e. coding of a motion vector MV consists of first building a motion vector predictor PMV for the vector to be coded and then transmitting the difference: DMV, where DMV=MV−PMV between the motion vector and the motion vector predictor. Rusert, ¶¶0011-0013: [0011] Accordingly, there is provided a method of selecting PMV candidates, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	previous block having a distance from a current block. The method comprises identifying allowed distance values of distances between the current block and the previous block. The method further comprises selecting a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. [0012] A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. [0013] Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. Rusert, ¶0014: At least one group of PMV candidates may have a common distance value. The method disclosed herein may also be used for managing PMV candidates. Rusert, ¶¶0015-0016: [0015] The method may further comprise identifying allowed positions of previous blocks, and wherein the selected set of PMV candidates comprises a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block and an allowed position.

Nokia Exhibit 2006, p. 56
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0016] The allowed positions may be are corner and middle block positions. A previous block at a corner position is offset from a current block position by an equal distance in a horizontal direction and a vertical direction. A previous block at a middle position is either horizontally aligned or vertically aligned with a current block position. In a system where coding is performed by horizontal lines of pixels starting in the top left corner, the allowed block positions may comprise: blocks horizontally aligned with and offset to the left of the current block; blocks vertically aligned with and offset above the current block; and blocks offset to the left and above the current block by the same distance. The allowed distance values and/or the allowed positions may be predetermined. Rusert, ¶0021: Any unnecessary PMV candidates may be removed from the set of PMV candidates. This ensures the length of the list is not unnecessarily long, which would reduce coding efficiency. A PMV candidate may be determined to be unnecessary if it at least one of the following conditions is fulfilled: the PMV candidate is a duplicate of another PMV candidate in the set; the PMV candidate is determined to be within a threshold distance of an existing PMV candidate; and the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits. Rusert, ¶¶0024-25: [0024] There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block.

Nokia Exhibit 2006, p. 57
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0025] There is further provided a video decoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0035: The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0062: A problem with this arrangement is that sometimes motion vectors close to the current block

Nokia Exhibit 2006, p. 58
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	are all exactly the same. It has been identified as advantageous to discard such duplicates. However, a problem with the scan pattern of FIG. 3 l is that a large number of the nearest neighbor backs must be investigated before any duplicates may be discarded. This means that a large number of vectors must sometimes be investigated before the candidate list can be filled and as such the above scan pattern can be considered to be slow. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance $(x_0-x_1)^2+(y_0-y_1)^2$ or absolute distance $\|x_0-x_1\|+\|y_0-y_1\|$, with (x_0,y_0) and (x_1,y_1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0090: Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful. Rusert, ¶0091-0094: [0091] As an example, assume we have the following candidates:

Nokia Exhibit 2006, p. 59
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

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

[0092] Also, assume that we encode a difference, DMV, (xdiff, ydiff) where xdiff and ydiff are encoded using Table 3 below. If we want to encode a motion vector, such as MV=(0,2), we can then encode it using candidate 3, which is PMV=(0,2), plus a difference DMV=(0,0):

PMV+DMV=MV

(0,2)+(0,0)=(0,2).

[0093] The index costs four bits (the code length for index 3 is four bits), and each of the zeroes in the difference cost one bit, so the total number of bits required to code MV=(0,2) is 4+1+1=6 bits.

[0094] However, we can also code the vector using index 0, PMV=(−1,2), plus a difference DMV=(1,0):

(−1,2)+(1,0)=(0,2).

Nokia Exhibit 2006, p. 60
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0095: The index costs one bit (the code length for index 1 is one bit), the xdiff=1 term in the difference costs three bits (see Table 3 below) and the ydiff=0 term costs one bit. Hence we get 1+3+1=5 bits in total, which is better than using index 3, which required 6 bits. It is easy to see given that the vector difference is coded using Table 3 below, that it will never be beneficial to use index 3, because using index 0 will always be one hit cheaper or better. Hence we can eliminate the candidate vector (0,2) and we get instead:

Value	Index	Code
(−1,2)	0	1
(13, 4)	1	010
(12, 3)	2	011
(3, 4)	4	001
(−4, 1)	5	0000
(4, 8)	6	0001

See also Rusert ¶¶ 0088-0098, reproduced earlier in limitation 1[a].

Rusert, ¶0074-0075:

[0074] Alternatively, removal of candidates from a PMV_CANDS list may be signaled explicitly from the encoder to the decoder (and thus decided dynamically by the encoder), e.g. by sending a code for removal of a motion vector candidate from a list along with an identifier of the motion vector, e.g. an index.

[0075] How to determine the order of motion vector candidates in candidate list will now be addressed. It is assumed that the candidates in PMV_CANDS are sorted, and use of one of the

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	candidates in PMV_CANDS for prediction is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. The following methods can be used when updating a PMV_CANDS list in order to sort the candidates in a way that is beneficial for overall coding efficiency. Rusert, ¶0084-0087: [0084] The candidate predictor list size can be varied. Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words. On the other hand, depending on video sequence characteristics, it may be beneficial to have a larger number of motion vector prediction candidates e.g. in order to save bits for DMV coding in case of irregular motion. The following methods can be used to adapt the size of PMV_CANDS list according to video sequence characteristics. [0085] The list size can be defined in the slice/frame/picture header or in a sequence-wide header (such as parameter set), i.e. signaled from the encoder to the decoder, and thus dynamically adapted by the encoder. [0086] Candidates that have not been used for prediction during encoding of a number of previously coded blocks (according to a pre-defined threshold) can be removed from the list, thus reducing the list size. [0087] The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased.

Nokia Exhibit 2006, p. 62
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0070: Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PUN is used for motion vector prediction, since shorter lists require shorter code words. Additionally, restricting the addition of certain candidates can make room for other, more beneficial candidates to be added. Rusert, ¶0012: A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. Rusert, ¶0064: Accordingly, a further embodiment is provided which can be said to speed up the scan pattern. In this embodiment entire layers of blocks having common Chebyshev distance are excluded from consideration. For instance, we could use only layers that are powers of two: 1, 2, 4, 8,.....This is an example of an exponential increase, but any series which grows greater than a linear increase will provide a good trade off between a selection of blocks close to the current block and a few blocks a significant distance away. Using Chebyshev distances of 1, 2, 4, 8 etc, the ordering in this embodiment is shown in FIG. 3 m. Rusert, ¶0065: However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded.

Nokia Exhibit 2006, p. 63
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. Rusert, ¶0048: as soon as a pre-defined number of unique PMV candidates have been found, Rusert, ¶0107: In one embodiment of the proposed solution, we use a maximum of four candidates in the list. However, in another embodiment, we use seven, and there is in principle no limit to the maximum. If we allow for a bigger maximum, the list can grow and chances increase that a suitable vector can be found. On the other hand, the number of bits needed to specify the candidate vector also increases. On top of that we get the problem that many vectors can be represented using several candidates, which is unnecessary. This redundant representation grows the more vectors are added. Rusert, ¶0113: FIG. 6 illustrates a method according to the present application. At 610 allowed distance values are identified for the distances between a current block and a previous block. The previous block may have a motion vector associated therewith which was used for coding said previous block. Optionally, at 620 allowed positions for previous blocks may be identified. Such allowed positions may comprise middle and corner positions in a square of blocks centered on the current block. At 630 a set of PMV candidates is selected as a subset of the previously used motion vectors associated with previous blocks having allowed distance values, and, if optional step 620 is implemented, allowed positions. The set of PMV candidates may then used for coding of the current block. *See also* Rusert, Figs. 3a-3n, reproduced earlier.

Nokia Exhibit 2006, p. 64
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶¶ 0036-0037: [0036] The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). [0037] The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. See also the disclosures cited for limitations 1[b], *supra.* *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*

Nokia Exhibit 2006, p. 65
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
1[d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and	Rusert, as evidenced by the example citations below, discloses determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing. *See, e.g.,* Rusert, ¶0012: A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. Rusert, ¶0020-0021: [0020] The method may further comprise moving unnecessary PMV candidates from the set of PMV candidates. [0021] Any unnecessary PMV candidates may be removed from the set of PMV candidates. This ensures the length of the list is not unnecessarily long, which would reduce coding efficiency. A PMV candidate may be determined to be unnecessary if it at least one of the following conditions is fulfilled: the PMV candidate is a duplicate of another PMV candidate in the set; the PMV candidate is determined to be within a threshold distance of an existing PMV candidate; and the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits. Rusert, ¶0090: Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list

Nokia Exhibit 2006, p. 66
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful. Rusert, ¶0022: Further, a set of PMV candidates may be determined to be unnecessary if removing them from the list of PMV candidates would result in at most N extra bits being required to encode any single motion vector, wherein N is a predetermined threshold Removing a set of PMV candidates from the list of PMV candidates allows the remaining PMV candidates to be signaled using shorter codes and so fewer bits. However, removing a set of PMV candidates from the list of PMV candidates will result in some motion vectors having a larger difference vector, which will require more bits to encode. Over the average of several motion vectors, the saving in signaling which PMV candidate to use can exceed the, at most, N extra bits required to signal some motion vectors. Rusert, ¶0039: Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive the list PMV_CANDS in the same way as the encoder.

Nokia Exhibit 2006, p. 67
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶0070: Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PUN is used for motion vector prediction, since shorter lists require shorter code words. Additionally, restricting the addition of certain candidates can make room for other, more beneficial candidates to be added. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list.

Nokia Exhibit 2006, p. 68
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0084: The candidate predictor list size can be varied. Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words. On the other hand, depending on video sequence characteristics, it may be beneficial to have a larger number of motion vector prediction candidates e.g. in order to save bits for DMV coding in case of irregular motion. The following methods can be used to adapt the size of PMV_CANDS list according to video sequence characteristics. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased.

Nokia Exhibit 2006, p. 69
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0050: > Note that sorting the list on distances may be avoided in an outwards going scan, by inserting unique vectors at the end of the PMV_CANDS list, the list is kept sorted with the spatially closest vector first in the list. See also the disclosures cited for limitations 1[c], *supra.* > To the extent this limitation is not expressly or inherently disclosed by Rusert, this limitation would have been obvious because Rusert teaches the advantages of reducing the number of candidates in PMV_CANDS using the three comparisons outlined in IPR2024-00604, Petition at 40-45 and IPR2024-00605, Petition at 37-43, as well as demonstrated by the disclosures set forth above in limitation 1[c]. *E.g.*, Rusert, ¶12, ¶21, ¶70, ¶84, ¶90. In addition, the natural time to perform such 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[.]" Rusert, ¶90. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*

Nokia Exhibit 2006, p. 70
ASUS v. Nokia
IPR2025-01153

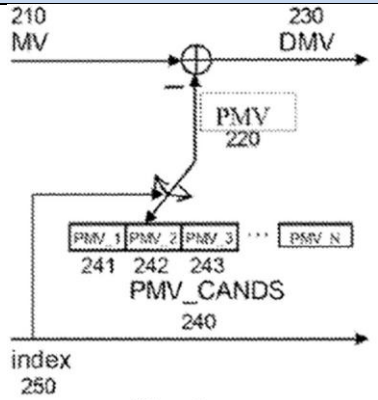
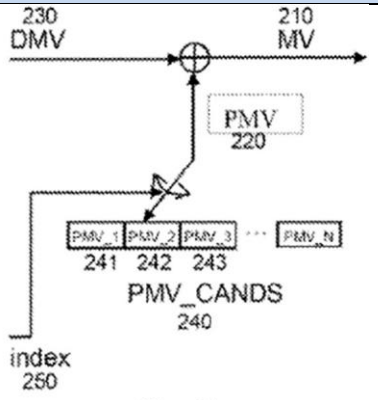
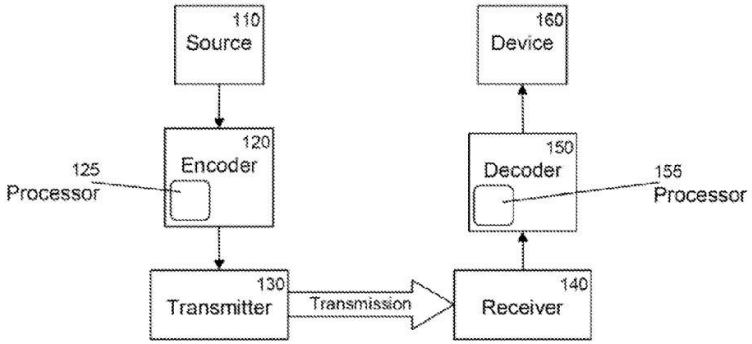
ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
1[e] causing information identifying the one spatial motion vector prediction candidate from the motion vector prediction list to be transmitted to a decoder or to be stored.	Rusert, as evidenced by the example citations below, discloses causing 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. *See, e.g.,* the disclosures cited for limitations 1[a], *supra.* Rusert, ¶0035: > The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶0036: > The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0037: > The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2

Nokia Exhibit 2006, p. 71
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	(242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. Rusert, ¶0023: The method may be for video encoding or for video decoding, wherein the current block is the block being encoded or decoded respectively. Rusert, ¶0027: An improved method and apparatus for selecting PMV candidates will now be described, by way of example only, with reference to the accompanying drawings, in which: Rusert, ¶0038: There are two major operations used for construction of the set of PMV candidates 240, initialization and update. Rusert, ¶0039: Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the list PMV_CANDS in the same way as the encoder. Rusert, ¶0091-0092, Figs. 2a, 2b: [0091] As an example, assume we have the following candidates: 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 [0092] Also, assume that we encode a difference, DMV, (xdiff, ydiff) where xdiff and ydiff are encoded using Table 3 below. If we want to encode a motion vector, such as MV=(0,2), we can then encode it using candidate 3, which is PMV=(0,2), plus a difference DMV=(0,0): PMV+DMV=MV (0,2)+(0,0)=(0,2).

Nokia Exhibit 2006, p. 73
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	 Fig. 2a  Fig. 2b Rusert, Fig. 1:  Fig. 1 Rusert, ¶0034:

Nokia Exhibit 2006, p. 74
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. Rusert, ¶0012: A PMV candidate list is created by selecting a subset of the motion vectors previously used for previous blocks. Restricting the previous blocks that are considered reduces the number of previous motion vectors that must be considered meaning that less computation is needed, improving the processor efficiency of the coding. Rusert, ¶¶0088-102, and related Tables, reproduced earlier for limitation 1[a], *supra*. Rusert, ¶0075: How to determine the order of motion vector candidates in candidate list will now be addressed. It is assumed that the candidates in PMV_CANDS are sorted, and use of one of the candidates in PMV_CANDS for prediction is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. The following methods can be used when updating a PMV_CANDS list in order to sort the candidates in a way that is beneficial for overall coding efficiency. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or*

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	more of any of the references identified in Exhibit D-10.
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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, Claim 1. *See, e.g.*, the disclosures cited in limitation 1[a], *supra*. Rusert, ¶0030: FIGS. 3 a to 3 n illustrate possible scan patterns and distance measurements for a plurality of blocks surrounding a current block. Rusert, ¶0040:

Nokia Exhibit 2006, p. 76
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Alternatively, one or more motion vector candidates that have not been encoded previously may be added into the PMV_CANDS list at the encoder, and then those motion vectors will be explicitly signaled to the decoder for use with PMV_CANDS, such that PMV_CANDS can be updated in the same way both at the encoder and the decoder. Rusert, ¶¶0051-0066: [0051] In one embodiment the ordering is based on Euclidian distance to the block in question. Assume for instance that the block we wanted to encode was in position x=3 y=5. Then we would rank the surrounding blocks according to euclidian_dist=sqrt((xpos−3)^2+(ypos−5)^2). [0052] For instance, a block in position (2,1) would have a distance sqrt(1^2+4^2) sqrt(17). Since the square-root does not change the order it can be excluded. Thus, if the block marked with a dot in FIG. 3 a is the one to be coded, the surrounding blocks would have the (squared) distances shown in FIG. 3 a (this diagram shows only the nearest neighbors and, in practice would be extended outwards to the edge of the frame. [0053] We then select blocks in an order that took the closest ones first, with some tie-breaking rule for blocks at the same distance. A possible order for the blocks shown in FIG. 3 a is shown in FIG. 3 b, where the order is shown according to: 1, 2, 3, . . . , 9, a, b, ... [0054] Of course some of these would never be available, such as “3” and “b” since they are directly to the right of the block to be coded, and would therefore be coded after the present block. Other blocks would sometimes be available depending upon the traversal pattern used. Some blocks may not be available due to the simple fact that there was no motion vector present in that block, or that the motion vector present was the same as a block earlier in the sequence. After these considerations it is possible that only a subset of blocks have valid motion vectors. An example of such a subset is shown in FIG. 3 c. In this case the list would consist of the motion vectors related to the shown blocks in the following order: 1, 2, 9, h, j, l, m.

Nokia Exhibit 2006, p. 77
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0055] A problem with this pattern is that it is quite hard to "spiral outwards", that is: it is not easy to calculate which x,y-coordinate a candidate with rank k has. Of course, they could be stored in a long list (since mapping is deterministic) but that list could be huge. [0056] Accordingly, an alternative embodiment is provided using an alternative structure for which it is easier to calculate the x,y-position of an element of a certain rank. This achieved by changing the distance metric from Euclidian distance to Chebyshev distance (also known as "box distance"). The new distance is instead calculated as Chebyshev_dist=max(|xpos−xcurrent|,|ypos−ycurrent|), [0057] In the example used above, where (xcurrent, ycurrent) (3, 5), a block in position (2,1) would have Chebyshev distance=4, since $\text{Chebyshev_dist} = \max(|2-3|, |1-5|)$ $= \max(|-1|, |-4|)$ $= \max(1, 4)$ $= 4.$ [0058] The surrounding blocks would have the Chebyshev distances shown in FIG. 3 d. This shows what looks like a square inside another square, or a box inside a box, hence the name box distance. Using Chebyshev distance it is a simple procedure to scan the candidates in order, for instance by first taking the top row, and then continuing anti-clockwise, as shown in FIG. 3 e. This illustrates a spiral scan, because the scan order spirals outwards. [0059] However, this ordering does have some problems. For instance, although blocks 3 and

Nokia Exhibit 2006, p. 78
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	4 have the same Chebyshev distance, block 4 has a lower Euclidian distance and so it would be appropriate to exchange the order of these two blocks. This problem is solved by using an order which is first determined by Chebyshev distance, and second by Euclidian distance. To clarify if two objects have different Chebyshev distance, the item with the smaller Chebyshev distance will always end up before the other one in the order. However, for items of the same Chebyshev distance, the order is determined by Euclidian distance. If two items have the same Chebyshev distance and the same Euclidian distance, a tie-breaking rule will be used. This provides an accurate ordering of the blocks that is still somewhat simple to calculate. An example of this ordering scheme will now be described. FIG. 3 f shows the Chebyshev distances of blocks up to three rows or columns from a central current block. As shown in FIG. 3 f, this gives several "layers" or squares of blocks with common Chebyshev distance. Further ordering can be provided by using a for-loop going over each layer (Chebyshev distance=1, 2, 3, etc.). In fact, since the blocks to the right and below the current block position have not been visited yet, and do not yet have motion vectors known for them, we can remove them from consideration. The resulting pattern is shown in FIG. 3 g. [0060] The remaining blocks are ordered as follows. Consider the layer of blocks with Chebyshev distance of 3. We then know that the items in the middle of each side of the layer will be the ones closest to the current block in a Euclidean sense, so we start by visiting them in some order, as shown in FIG. 3 h. The second closest blocks will be the ones right next to these, so we proceed to next apply an order to these as shown in FIG. 3 i. The next closest blocks will be those next to the ones just ordered as shown in FIG. 3 j. Finally, the three corner blocks are placed in the order as shown in FIG. 3 k. [0061] We have now a finished layer 3, and would need to proceed to layer 4 if we had not found the required number of different motion vectors. A complete ordering for the blocks up to Chebyshev distance 4 is shown in FIG. 3 l, where the order is shown as 1, 2, . . . 9, a, b, . . . z, A, B, . . . U. [0062] A problem with this arrangement is that sometimes motion vectors close to the current block are all exactly the same. It has been identified as advantageous to discard such

Nokia Exhibit 2006, p. 79
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	duplicates. However, a problem with the scan pattern of FIG. 3 l is that a large number of the nearest neighbor backs must be investigated before any duplicates may be discarded. This means that a large number of vectors must sometimes be investigated before the candidate list can be filled and as such the above scan pattern can be considered to be slow. [0063] Another disadvantage with the scan pattern of FIG. 3 l is that motion vectors close to each other may be similar to each other even if they are not exactly the same. Thus the resulting list of PMVs may lack the diversity necessary encode motion vectors efficiently. For instance, the current block may be part of the background whereas all MVs 1, 2, . . . , 9, a, b, . . . , z, A, B, . . . , U may be part of a moving object. It could then be advantageous to be able to reach MVs corresponding to b her away (e.g., in layer 4, 5 etc) before filling up the list. [0064] Accordingly, a further embodiment is provided which can be said to speed up the scan pattern. In this embodiment entire layers of blocks having common Chebyshev distance are excluded from consideration. For instance, we could use only layers that are powers of two: 1, 2, 4, 8,.....This is an example of an exponential increase, but any series which grows greater than a linear increase will provide a good trade off between a selection of blocks close to the current block and a few blocks a significant distance away. Using Chebyshev distances of 1, 2, 4, 8 etc, the ordering in this embodiment is shown in FIG. 3 m. [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance.

Nokia Exhibit 2006, p. 80
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, Figs. 3a-3n, reproduced *supra*. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* Claim 1. *See, e.g.,* the disclosures cited in limitation 1[c], *supra*. Rusert, ¶0021: > Any unnecessary PMV candidates may be removed from the set of PMV candidates. This ensures the length of the list is not unnecessarily long, which would reduce coding efficiency. A PMV candidate may be determined to be unnecessary if it at least one of the following conditions is fulfilled: the PMV candidate is a duplicate of another PMV candidate in the set; the PMV candidate is determined to be within a threshold distance of an existing PMV

Nokia Exhibit 2006, p. 81
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	candidate; and the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits. Rusert, ¶¶0038-0040: [0038] There are two major operations used for construction of the set of PMV candidates 240, initialization and update. [0039] Initialization means that a certain pre-defined state of the list is established. A PMV_CANDS list may be initialized e.g. as an empty list (zero entries) or as a list with one or more pre-defined entries such as the zero vector (0,0). Update means that one or more motion vectors are added to an existing PMV_SANDS list. A PMV_CANDS list may be updated to include previously coded motion vectors MV. At the encoder, when encoding a current block, PMV_CANDS may contain, besides any pre-defined initialization vectors, motion vectors associated with previously encoded blocks in the video. By restricting the possible candidates in PMV_CANDS to pre-defined vectors and previously coded vectors, the decoder can derive the list PMV_CANDS in the same way as the encoder. [0040] Alternatively, one or more motion vector candidates that have not been encoded previously may be added into the PMV_CANDS list at the encoder, and then those motion vectors will be explicitly signaled to the decoder for use with PMV_CANDS, such that PMV_CANDS can be updated in the same way both at the encoder and the decoder. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions,

Nokia Exhibit 2006, p. 82
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶¶0051-66, reproduced *supra*. Rusert, Figs. 3a-3n, reproduced *supra*. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*

Nokia Exhibit 2006, p. 83
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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, as evidenced by the example citations below, discloses 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. See, e.g., Rusert, ¶¶0034-36: [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. [0035] The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. [0036] The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code “index” 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV

Nokia Exhibit 2006, p. 84
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶¶0024-35: [0024] There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. [0025] There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. [0026] There is further provided a computer-readable medium, carrying instructions, which, when executed by computer logic, causes said computer logic to carry out any of the methods disclosed herein. [0027] An improved method and apparatus for selecting PMV candidates will now be described, by way of example only, with reference to the accompanying drawings, in which: [0028] FIG. 1 shows a video coding and transmission system;

Nokia Exhibit 2006, p. 85
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0029] FIGS. 2 a and 2 b show the use of a PMV candidate list during encoding and decoding respectively; [0030] FIGS. 3 a to 3 n illustrate possible scan patterns and distance measurements for a plurality of blocks surrounding a current block; [0031] FIG. 4 shows an example of two PMV candidates; [0032] FIG. 5 shows the bit cost of coding motion vectors using the example motion vectors of FIG. 4; and [0033] FIG. 6 illustrates a method disclosed herein. [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. [0035] The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances.

Nokia Exhibit 2006, p. 86
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, Fig. 1, reproduced supra. See also the disclosures of limitations 1[a] and 1[e], supra. The disclosures of Karczewicz are also relevant to this limitation: Karczewicz, ¶0050: > In this disclosure, the term "coding" refers to encoding or decoding. Similarly, the term "coder" generally refers to any video encoder, video decoder, or combined encoder/decoder (codec). Accordingly, the term "coder" is used herein to refer to a specialized computer device or apparatus that performs video encoding or video decoding. The CBP coding techniques of this disclosure may be applicable to encoders or decoders. The encoder uses the described VLC techniques to encode the CBP, and the decoder uses reciprocal VLC techniques to decode the CBP. To the extent this limitation is not expressly or inherently disclosed by Rusert, it is disclosed by Lin. See, e.g., Lin, ¶0004: > High-Efficiency Video Coding (HEVC) is a new international video coding standard that is being developed by the Joint Collaborative Team on Video Coding (JCT-VC). HEVC is based on the hybrid block-based motion-compensated DCT-like transform coding architecture. The basic unit for compression, termed Coding Unit (CU), is a 2N×2N square block, and each CU can be recursively split into four smaller CUs until a predefined minimum size is reached. Each CU contains one or multiple Prediction Units (PUs). The PU sizes can be 2N×2N, 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N, nR×2N, or N×N, where 2N×N, 2N×nU, 2N×nD and N×2N, nL×2N, nR×2N correspond to horizontal and vertical partition of a 2N×2N PU with symmetric or asymmetric PU size division respectively. Lin, ¶0006:

Nokia Exhibit 2006, p. 87
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The Skip and Merge modes utilize motion inference methods (MV=MVP+MVD where MVD is zero) to obtain the motion information from spatial neighboring blocks (spatial candidates) or temporal blocks (temporal candidates) located in a co-located picture. The co-located picture is the first reference picture in list 0 or list 1, which is signaled in the slice header. Lin, ¶0025: A method and apparatus of deriving a motion vector predictor (MVP) for a current block in an Inter, Merge, or Skip mode are disclosed. Embodiments according to the present invention determine redundant MVP candidates according to a non-MV-value based criterion. The redundant MVP candidates are then removed from the MVP candidate set. In one embodiment, MVP candidates within a same PU (Processing Unit) or same MDSR (Motion Data Storage Reduction) unit are determined to be redundant. MVP candidates, that located in different PUs and the different PUs are merged, are also determined to be redundant. In another embodiment, one MVP candidate is redundant if a first MVP candidate is coded as an Inter mode according to a second MVP candidate and zero MVD (Motion Vector Difference) and without scaling. In yet another embodiment, one or more of the MVP candidates are redundant if said one or more of the MVP candidates cause the second 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU to be merged into a 2N×2N PU. Similarly, one or more of the MVP candidates are determined to be redundant if said one or more of the MVP candidates cause the fourth N×N PU to be merged into a 2N×N, N×2N, 2N×nU, 2N×nD, nL×2N, nR×2N or 2N×2N PU. In another embodiment, for the second 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N 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) 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU. Lin, ¶0044: One embodiment of the present invention identifies and removes redundant MVP candidates to avoid imitation of Merge. In HEVC, a 2N×2N PU may be partitioned into 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs according to the Rate-Distortion Optimization (RDO)

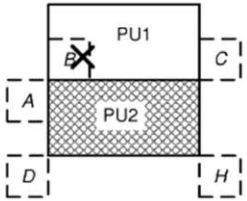
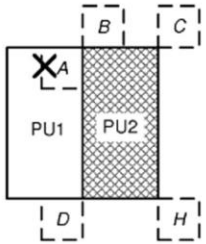
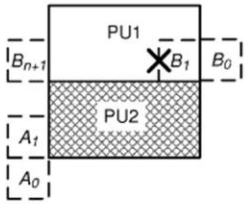
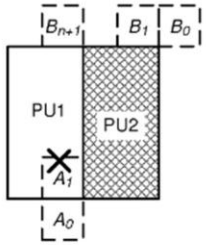
Nokia Exhibit 2006, p. 88
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	process, where 2N×nU, 2N×nD, nL×2N and nR×2N are asymmetrical motion partition (AMP) PUs. If the second 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU is merged with the first PU, the two 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs will be merged back into one 2N×2N PU. Therefore, the second PU should avoid the respective 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N Merge mode so that two 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs will not be merged back into a 2N×2N PU. The scenario that multiple partitioned PUs from an original PU are merged back to form the original PU is termed imitation of Merge. Consequently, the MVP candidates of a current PU that may cause the current PU to be merged with other PUs to cause imitation of Merge is considered redundant and can be removed without comparing the MV values. For example, for the second PU of 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N Merge mode, those MVPs which are located within the previous (first) PUs and will naturally make this 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU merge as a 2N×2N PU can be removed without comparing the values of MVs. FIG. 7A and FIG. 7B illustrate examples of redundancy due to imitation of 2N×N and N×2N Merge respectively. As shown in FIG. 7A, MVP B for the second 2N×N PU in the 2N×N Merge mode can be removed to avoid the duplication. In FIG. 7A, MVP A for the second N×2N PU in the N×2N Merge mode can be removed to avoid the imitation of Merge. The neighboring block configuration shown in FIGS. 7A and 7B for determining spatial/temporal MVPs are used for illustration purpose. A system may adopt other neighboring block configurations, and the MVP redundancy due to the imitation of Merge can be identified and removed according to the present invention. For example, the neighboring block configuration for the Inter and Skip modes as shown in FIG. 1 may be used for the Merge mode as well. The redundancy due to imitation of Merge can be determined as shown in FIG. 7C-FIG. 7H. In FIG. 7C, MVP B1 for 2N×N PU2 is from 2N×N PU1. The 2N×N PU1 and PU2 merging will cause the imitation of Merge. Accordingly, MVP B1 is determined to be redundant and is removed from the MVP candidate set. Similarly, MVP A1 in FIG. 7D, MVP B1 in FIG. 7E, MVP A1 in FIG. 7F, MVP B1 in FIG. 7G and MVP A1 in FIG. 7H are determined to be redundant and are removed from the MVP candidate set, respectively. Lin, ¶0047:

Nokia Exhibit 2006, p. 89
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	One embodiment of the present invention identifies and removes redundant MVP candidates using motion ID. This embodiment utilizes motion ID to track the inheritance process of motion information passed from a coded PU to the current PU so as to follow the trail of the MV. The motion ID is used to help identify redundancy in MVP candidates without the need for comparing the MV values. In addition, motion ID also offers a benefit of alleviating parsing error and error propagation. In order to reduce the bits used to encode the index of motion vector predictor (MVP), redundant MVPs are removed from the MVP candidate set. However, this can cause parsing errors leading to severe error propagation. When a motion vector (MV) of a previous picture cannot be decoded correctly, a mismatch between the candidate set on the encoder side and that on the decoder side may occur, resulting in parsing error of the index of the best MVP. Consequently, the rest of the current picture cannot be parsed or decoded properly. What is even worse, this parsing error can affect subsequent inter pictures that also use temporal MVPs. Motion identification (ID) is proposed to describe the inheritance process of motion information passed from a coded PU to the current PU to follow the trail of the MV. A unique motion ID is assigned for each new MV created in the Inter mode and can be inherited through the Merge and Skip modes. By comparing the motion ID, the MVP having the same motion ID as a previous MVP in the candidate set can thus be removed. Even if the MV of a previous picture had not been correctly decoded, the decision of redundancy will not be affected because the inherent of motion ID does not rely on the motion vector values. Lin, Fig.7A-7D:

Nokia Exhibit 2006, p. 90
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	 Fig. 7A  Fig. 7B  Fig. 7C  Fig. 7D See also Lin Provisional at 7, 8-11, 17 (corroborating Lin's disclosures) Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.
5[preamble] The method according to claim 1, further	*See* claim 1.

Nokia Exhibit 2006, p. 91
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
comprising	
5[a] determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and	Rusert, as evidenced by the example citations below, discloses determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list. *See, e.g.,* Rusert, ¶0013: > Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. Rusert, ¶0070: > Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PUN is used for motion vector prediction, since shorter lists require shorter code words. Additionally, restricting the addition of certain candidates can make room for other, more beneficial candidates to be added. Rusert, ¶0073: > Also, the number of candidates in PMV_CANDS may be limited to a pre-defined or dynamically obtained number. It is possible that once the number has been reached an additional candidate is to be added, then the candidate at the end of the PMV_CANDS list may be removed. This can be done because the candidate list is sorted such that the PMV candidate at the end of the list has been determined to be the least likely to be used.

Nokia Exhibit 2006, p. 92
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0077: The motion vector associated with the last coded block is placed at the beginning of the list (shortest code word). Alternatively, a combined candidate such as an H.264/AVC median predictor (or the like) for the current block is placed at the beginning of the list. Combining this approach with dynamic adaptation of PMV_CANDS list size allows guaranteed prediction performance of e.g. the H.264/AVC median predictor, since it is possible that PMV_CANDS list size is set to one, such that no bits need to be sent for index signaling. Rusert, ¶¶0084-0087: The candidate predictor list size can be varied. Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words. On the other hand, depending on video sequence characteristics, it may be beneficial to have a larger number of motion vector prediction candidates e.g. in order to save bits for DMV coding in case of irregular motion. The following methods can be used to adapt the size of PMV_CANDS list according to video sequence characteristics. The list size can be defined in the slice/frame/picture header or in a sequence-wide header (such as parameter set), i.e. signaled from the encoder to the decoder, and thus dynamically adapted by the encoder. Candidates that have not been used for prediction during encoding of a number of previously coded blocks (according to a pre-defined threshold) can be removed from the list, thus reducing the list size. The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-

Nokia Exhibit 2006, p. 93
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. Rusert, ¶¶0088-0090: As mentioned above, the candidates in PMV_CANDS are sorted and the use of one of the candidates in PMV_CANDS is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. Such code words can be defined e.g. according to Variable Length Coding (VLC) tables. The VLC table used can depend on the maximum number of candidates in PMV_SANDS (the list size), as e.g. dynamically adapted according to the methods above. Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list.

Nokia Exhibit 2006, p. 94
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 1

Example VLC codes for different maximum list sizes.

Maximum list size C	Index: VLC code
	0: - (unambiguous, no signaling necessary)
2	0: 0
	1: 1
3	0: 1
	1: 00
	2: 01
4	0: 1
	1: 00
	2: 010
	3: 011
5	0: 1
	1: 00
	2: 010
	3: 0110
	4: 0111
6	0: 1
	1: 010
	2: 011
	3: 001
	4: 0000
	5: 0001
7	0: 1
	1: 010
	2: 011
	3: 0010
	4: 0011
	5: 0000
	6: 0001

Nokia Exhibit 2006, p. 95
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 2

VLC code for coding of two candidates indexes associated with bi-predicted block, C: maximum size of PMV_CANDS.

Case	VLC code
idx1 < idx0	VLC0 (idx0,C) 0 VLC0 (idx1,C-1)
idx1 = idx0	VLC0 (idx0,C) 1
idx1 > idx0	VLC0 (idx0,C) 0 VLC0 (idx1-1,C-1)

For bi-predicted motion compensation blocks, two motion vectors are coded, and thus two PMV candidates can be necessary. In that case the index numbers for the two PMV candidates can be coded together to further reduce the number of bits required for index coding. Table 2 shows an example for joint index coding, considering that both motion vectors use the same PMV_CANDS list, and that it is likely that both motion vectors use the same predictor in the PMV_CANDS list. Here idx0 and idx1 denote the indexes for first and second predictor, respectively. VLC0(idx,C) denotes a VLC for an index "idx" according to Table 1 considering a maximum list size of C.

Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful.

Nokia Exhibit 2006, p. 96
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0107: In one embodiment of the proposed solution, we use a maximum of four candidates in the list. However, in another embodiment, we use seven, and there is in principle no limit to the maximum. If we allow for a bigger maximum, the list can grow and chances increase that a suitable vector can be found. On the other hand, the number of bits needed to specify the candidate vector also increases. On top of that we get the problem that many vectors can be represented using several candidates, which is unnecessary. This redundant representation grows the more vectors are added. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
5[b] limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.	Rusert, as evidenced by the example citations below, discloses limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number. *See, e.g.,* Rusert, ¶¶0044-0049: [0044] The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞

Nokia Exhibit 2006, p. 97
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: [0045] all blocks of the current frame have been scanned, [0046] all blocks of a pre-defined number of subsequent frames (e.g. the last frame) have been scanned [0047] once a certain distance has been reached, [0048] as soon as a pre-defined number of unique PMV candidates have been found, [0049] all blocks of a predetermined scan pattern have been scanned. Rusert, ¶0073: Also, the number of candidates in PMV_CANDS may be limited to a pre-defined or dynamically obtained number. It is possible that once the number has been reached an additional candidate is to be added, then the candidate at the end of the PMV_CANDS list may be removed. This can be done because the candidate list is sorted such that the PMV candidate at the end of the list has been determined to be the least likely to be used. Rusert, ¶0088, Table 1: As mentioned above, the candidates in PMV_CANDS are sorted and the use of one of the candidates in PMV_CANDS is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. Such code words can be defined e.g. according to Variable Length Coding (VLC) tables. The VLC table used can depend on the

Nokia Exhibit 2006, p. 98
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	maximum number of candidates in PMV_SANDS (the list size), as e.g. dynamically adapted according to the methods above. Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list.

Nokia Exhibit 2006, p. 99
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 1

Example VLC codes for different maximum list sizes.

Maximum list size C	Index: VLC code
	0: - (unambiguous, no signaling necessary)
2	0: 0
	1: 1
3	0: 1
	1: 00
	2: 01
4	0: 1
	1: 00
	2: 010
	3: 011
5	0: 1
	1: 00
	2: 010
	3: 0110
	4: 0111
6	0: 1
	1: 010
	2: 011
	3: 001
	4: 0000
	5: 0001
7	0: 1
	1: 010
	2: 011
	3: 0010
	4: 0011
	5: 0000
	6: 0001

Further, to the extent Complainant contends this element is not expressly or inherently disclosed by

Nokia Exhibit 2006, p. 100
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.
6[preamble] The method according to claim 5 comprising:	*See* Claim 5.
6[a] examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;	Rusert, as evidenced by the example citations below, discloses examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number. *See, e.g.,* Rusert, ¶0013: > Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. Rusert, ¶0044-0049: > [0044] The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: [0045] All blocks of the current frame have been scanned, [0046] all blocks of a pre-defined number of subsequent frames (e.g. the last frame) have been scanned [0047] once a certain distance has been reached, [0048] as soon as a pre-defined number of unique PMV candidates have been found, [0049] all blocks of a predetermined scan pattern have been scanned. Rusert, ¶¶0070-77: [0070] Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PUN is used for motion vector prediction, since shorter lists require shorter code words. Additionally, restricting the addition of certain candidates can make room for other, more beneficial candidates to be added. [0071] One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list.

Nokia Exhibit 2006, p. 102
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0072] Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. [0073] Also, the number of candidates in PMV_CANDS may be limited to a pre-defined or dynamically obtained number. It is possible that once the number has been reached an additional candidate is to be added, then the candidate at the end of the PMV_CANDS list may be removed. This can be done because the candidate list is sorted such that the PMV candidate at the end of the list has been determined to be the least likely to be used. [0074] Alternatively, removal of candidates from a PMV_CANDS list may be signaled explicitly from the encoder to the decoder (and thus decided dynamically by the encoder), e.g. by sending a code for removal of a motion vector candidate from a list along with an identifier of the motion vector, e.g. an index. [0075] How to determine the order of motion vector candidates in candidate list will now be addressed. It is assumed that the candidates in PMV_CANDS are sorted, and use of one of the candidates in PMV_CANDS for prediction is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. The following methods can be used when updating a PMV_CANDS list in order to sort the candidates in a way that is beneficial for overall coding efficiency. [0076] The motion vectors corresponding to blocks that are close to the current block (using some distance metric) will get a position closer to the start of the list compared to motion vectors belonging to blocks that are further away from the current block. [0077] The motion vector associated with the last coded block is placed at the beginning of

Nokia Exhibit 2006, p. 103
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the list (shortest code word). Alternatively, a combined candidate such as an H.264/AVC median predictor (or the like) for the current block is placed at the beginning of the list. Combining this approach with dynamic adaptation of PMV_CANDS list size allows guaranteed prediction performance of e.g. the H.264/AVC median predictor, since it is possible that PMV_CANDS list size is set to one, such that no bits need to be sent for index signaling. Rusert, ¶¶0084-0090: [0084] The candidate predictor list size can be varied. Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words. On the other hand, depending on video sequence characteristics, it may be beneficial to have a larger number of motion vector prediction candidates e.g. in order to save bits for DMV coding in case of irregular motion. The following methods can be used to adapt the size of PMV_CANDS list according to video sequence characteristics. [0085] The list size can be defined in the slice/frame/picture header or in a sequence-wide header (such as parameter set), i.e. signaled from the encoder to the decoder, and thus dynamically adapted by the encoder. [0086] Candidates that have not been used for prediction during encoding of a number of previously coded blocks (according to a pre-defined threshold) can be removed from the list, thus reducing the list size. [0087] The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased.

[0088] As mentioned above, the candidates in PMV_CANDS are sorted and the use of one of the candidates in PMV_CANDS is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. Such code words can be defined e.g. according to Variable Length Coding (VLC) tables. The VLC table used can depend on the maximum number of candidates in PMV_SANDS (the list size), as e.g. dynamically adapted according to the methods above. Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list.

TABLE 1

Example VLC codes for different maximum list sizes.

Maximum list size C	Index: VLC code
	0: - (unambiguous, no signaling necessary)
2	0: 0
	1: 1
3	0: 1
	1: 00
	2: 01
4	0: 1
	1: 00
	2: 010
	3: 011
5	0: 1
	1: 00
	2: 010
	3: 0110
	4: 0111

Nokia Exhibit 2006, p. 105
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

6	0: 1 1: 010 2: 011 3: 001 4: 0000 5: 0001
7	0: 1 1: 010 2: 011 3: 0010 4: 0011 5: 0000 6: 0001

[0089] For bi-predicted motion compensation blocks, two motion vectors are coded, and thus two PMV candidates can be necessary. In that case the index numbers for the two PMV candidates can be coded together to further reduce the number of bits required for index coding. Table 2 shows an example for joint index coding, considering that both motion vectors use the same PMV_CANDS list, and that it is likely that both motion vectors use the same predictor in the PMV_CANDS list. Here idx0 and idx1 denote the indexes for first and second predictor, respectively. VLC0(idx,C) denotes a VLC for an index "idx" according to Table 1 considering a maximum list size of C.

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 2

VLC code for coding of two candidates indexes associated with bi-predicted block, C: maximum size of PMV_CANDS.

Case	VLC code
idx1 < idx0	VLC0 (idx0,C) 0 VLC0 (idx1,C-1)
idx1 = idx0	VLC0 (idx0,C) 1
idx1 > idx0	VLC0 (idx0,C) 0 VLC0 (idx1-1,C-1)

[0090] Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful.

Rusert, ¶0107:

In one embodiment of the proposed solution, we use a maximum of four candidates in the list. However, in another embodiment, we use seven, and there is in principle no limit to the maximum. If we allow for a bigger maximum, the list can grow and chances increase that a suitable vector can be found. On the other hand, the number of bits needed to specify the candidate vector also increases. On top of that we get the problem that many vectors can be

Nokia Exhibit 2006, p. 107
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	represented using several candidates, which is unnecessary. This redundant representation grows the more vectors are added. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[b] if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;	Rusert, as evidenced by the example citations below, discloses examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction. *See, e.g.,* Rusert, ¶¶0044-49, reproduced *supra*. Rusert, ¶¶0050-66, reproduced *supra*. Rusert, ¶0054: Of course some of these would never be available, such as "3" and "b" since they are directly to the right of the block to be coded, and would therefore be coded after the present block. Other blocks would sometimes be available depending upon the traversal pattern used. Some blocks may not be available due to the simple fact that there was no motion vector present in that block, or that the motion vector present was the same as a block earlier in the sequence. After these considerations it is possible that only a subset of blocks have valid motion vectors. An example of such a subset is shown in FIG. 3 c. In this case the list would consist of the motion vectors related to the shown blocks in the following order: 1, 2, 9, h, j, l, m. Rusert, Figs. 3a-3n, reproduced *supra*. Rusert, ¶¶0002-5: [0002] Recent video coding standards are based on the hybrid coding principle, which

Nokia Exhibit 2006, p. 108
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	comprises motion compensated temporal prediction of video frames and coding of frame residual signals. For efficient motion compensated temporal prediction, block-based motion models are used to describe the motion of pixel blocks across frames. Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames). These motion vectors are coded in the video bit stream along with the frame residual signals. [0003] At high compression ratios (or equivalently, low video bitrates), motion vector coding takes a large part of the total amount of bits, especially in recent video coding standards such as H.264/AVC where small motion compensation block sizes are used. Typically, lossless predictive coding of motion vectors is used, i.e. coding of a motion vector MV consists of first building a motion vector predictor PMV for the vector to be coded and then transmitting the difference: DMV, where DMV=MV−PMV between the motion vector and the motion vector predictor. [0004] In H.264/AVG, a PMV is derived as the median of the motion vectors of three spatially neighboring blocks. Other approaches consider also temporally neighboring blocks (i.e. co-located in neighboring frames) for motion vector prediction. Instead of using a fixed rule for building PMV, recently approaches have been presented that explicitly signal a PMV to be used out of a set of PMV candidates, PMV_CANDS. Although this requires additional bits to signal one candidate out of the set, it can overall save bits for motion vector coding, since DMV coding can be more efficient. That is, identifying a PMV and signaling DMV can take fewer bits than independently signaling the MV. [0005] The efficiency of motion vector coding schemes which use PMV candidate signaling depends on the suitability of the available candidates in PMV_CANDS. That is, the construction of the candidate list has a major impact on the coding performance. Existing approaches typically use motion vectors from spatially surrounding blocks or temporally neighboring blocks (co-located blocks in neighboring frames). Such construction of PMV_CANDS, i.e. considering only the few surrounding blocks as a source of motion vector

Nokia Exhibit 2006, p. 109
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	predictors, can be sub-optimal. Rusert, ¶0036: The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0043: Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks. *See also* the disclosures in limitation 1[a], reproduced *supra*. The disclosures of Karczewicz are also relevant to this limitation. *See, e.g.*, Karczewicz, ¶¶ 33-36: [0033] HM refers to a block of video data as a coding unit (CU). Syntax data within a bitstream may define a largest coding unit (LCU), which is a largest coding unit in terms of the number of pixels. In general, a CU has a similar purpose to a macroblock of H.264, except that a CU does not have the same size limitations associated with macroblocks. For example, a CU may be split into sub-CUs, and the size of the LCU may be different depending on the

Nokia Exhibit 2006, p. 110
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	scenario or situation. In general, references in this disclosure to a CU may refer to a LCU of a picture or a sub-CU of an LCU. A LCU may be split into sub-CUs, and each sub-CU may be split into sub-CUs. Syntax data for a bitstream may define the size of the LCU as well as a maximum number of times an LCU may be split, referred to as maximum CU depth. Accordingly, a bitstream may also define a smallest coding unit (SCU). This disclosure also uses the term "block" to refer to any of an LCU, SCU, CU, prediction unit (PU), or transform unit (TU) (PU and TU are described in further detail below). [0034] A LCU may be associated with a quadtree data structure. In general, a quadtree data structure includes one node per CU, where a root node corresponds to the LCU. If a CU is split into four sub-CUs, the node corresponding to the CU includes four child nodes, each of which corresponds to one of the sub-CUs or one or more sub-CUs of a sub-CU. Each node of the quadtree data structure may provide syntax data for the corresponding CU. For example, a node in the quadtree may include a split flag, indicating whether the CU corresponding to the node is split into sub-CUs. Syntax elements for a CU may be defined recursively, and may depend on whether the CU is split into sub-CUs. Additional syntax elements may also be included for each CU, such as flags indicating whether filtering should be applied to the CU. [0035] A CU that is not split may include one or more prediction units (PUs). In general, a PU represents all or a portion of the corresponding CU, and includes data for retrieving a reference sample for the PU. For example, when the PU is intra-mode encoded, the PU may include data describing an intra-prediction mode for the PU. As another example, when the PU is inter-mode encoded, the PU may include data defining a motion vector for the PU. The data defining the motion vector may describe, for example, a horizontal component of the motion vector, a vertical component of the motion vector, a resolution for the motion vector (e.g., one-quarter pixel precision or one-eighth pixel precision), a reference frame to which the motion vector points, and/or a reference list (e.g., list 0 or list 1) for the motion vector. Data for the CU defining the PU(s) may also describe, for example, partitioning of the CU into one or more PUs. Partitioning modes may differ between whether the CU is intra-prediction mode encoded, or inter-prediction mode encoded. [0036] A CU may also include one or more transform units (TUs) that may be used to perform transform operations on video data. Following prediction using a PU, a video encoder may calculate residual values for the portion of the CU corresponding to the PU. The

Nokia Exhibit 2006, p. 111
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	residual values may be transformed, scanned, and quantized. A TU is not necessarily limited to the size of a PU. Thus, TUs may be larger or smaller than corresponding PUs for the same CU. In some examples, the maximum size of a TU may correspond to the size of the corresponding CU. Karczewicz, ¶¶ 64, 66: [0064] In this disclosure, the phrase "block" refers to any size or type of video block. For example, the phrase "block" may refer to one or more of a macroblock, LCU, CU, sub-CU, TU, or PU. Moreover, blocks may refer to blocks of video data in the pixel domain, or blocks of data in a transform domain such as a discrete cosine transform (DCT) domain, a domain similar to DCT, a wavelet domain, or the like. In addition, the phrase block may refer to the luma and chroma blocks that are generally residual blocks of data in the transform domain. [0066] After generating the predictive block, the differences between the current video block being coded and the predictive block are coded as a residual block, and prediction syntax (such as a motion vector) is used to identify the predictive block. The residual block may be transformed and quantized. Transform techniques may comprise a DCT process or conceptually similar process, integer transforms, wavelet transforms, or other types of transforms. In a DCT or DCT-like process, as an example, the transform process converts a set of pixel values (e.g., residual values) into transform coefficients, which may represent the energy of the pixel values in the frequency domain. Quantization is typically applied on the transform coefficients, and generally involves a process that limits the number of bits associated with any given transform coefficient. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c] if so, performing at least one of the following:	Rusert, as evidenced by the example citations below, discloses this limitation, including performing: 6[c.4] for a potential spatial motion vector prediction candidate, which is below the potential spatial

Nokia Exhibit 2006, p. 112
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	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. *See, e.g.*, the disclosures set forth in limitations 1[a], reproduced *supra* and 6[c.4], reproduced *infra*. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n.

Nokia Exhibit 2006, p. 113
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute

Nokia Exhibit 2006, p. 114
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	distance, with a pre- defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.1] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one

Nokia Exhibit 2006, p. 115
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between

Nokia Exhibit 2006, p. 116
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.1.1] the received block of pixels is vertically divided into a first prediction unit and a second prediction unit;	Rusert, as evidenced by the example citations below, discloses the received block of pixels is vertically divided into a first prediction unit and a second prediction unit. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or

Nokia Exhibit 2006, p. 117
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for motion vectors that are similar but not equal, such as pairs of motion vectors that have a similarity measure smaller

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	than a pre-defined threshold, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. The disclosures of Karczewicz are also relevant to this limitation. *See, e.g.*, Karczewicz, ¶¶ 33-36: [0033] HM refers to a block of video data as a coding unit (CU). Syntax data within a bitstream may define a largest coding unit (LCU), which is a largest coding unit in terms of the number of pixels. In general, a CU has a similar purpose to a macroblock of H.264, except that a CU does not have the same size limitations associated with macroblocks. For example, a CU may be split into sub-CUs, and the size of the LCU may be different depending on the scenario or situation. In general, references in this disclosure to a CU may refer to a LCU of a picture or a sub-CU of an LCU. A LCU may be split into sub-CUs, and each sub-CU may be split into sub-CUs. Syntax data for a bitstream may define the size of the LCU as well as a maximum number of times an LCU may be split, referred to as maximum CU depth.

Nokia Exhibit 2006, p. 119
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Accordingly, a bitstream may also define a smallest coding unit (SCU). This disclosure also uses the term "block" to refer to any of an LCU, SCU, CU, prediction unit (PU), or transform unit (TU) (PU and TU are described in further detail below). [0034] A LCU may be associated with a quadtree data structure. In general, a quadtree data structure includes one node per CU, where a root node corresponds to the LCU. If a CU is split into four sub-CUs, the node corresponding to the CU includes four child nodes, each of which corresponds to one of the sub-CUs or one or more sub-CUs of a sub-CU. Each node of the quadtree data structure may provide syntax data for the corresponding CU. For example, a node in the quadtree may include a split flag, indicating whether the CU corresponding to the node is split into sub-CUs. Syntax elements for a CU may be defined recursively, and may depend on whether the CU is split into sub-CUs. Additional syntax elements may also be included for each CU, such as flags indicating whether filtering should be applied to the CU. [0035] A CU that is not split may include one or more prediction units (PUs). In general, a PU represents all or a portion of the corresponding CU, and includes data for retrieving a reference sample for the PU. For example, when the PU is intra-mode encoded, the PU may include data describing an intra-prediction mode for the PU. As another example, when the PU is inter-mode encoded, the PU may include data defining a motion vector for the PU. The data defining the motion vector may describe, for example, a horizontal component of the motion vector, a vertical component of the motion vector, a resolution for the motion vector (e.g., one-quarter pixel precision or one-eighth pixel precision), a reference frame to which the motion vector points, and/or a reference list (e.g., list 0 or list 1) for the motion vector. Data for the CU defining the PU(s) may also describe, for example, partitioning of the CU into one or more PUs. Partitioning modes may differ between whether the CU is intra-prediction mode encoded, or inter-prediction mode encoded. [0036] A CU may also include one or more transform units (TUs) that may be used to perform transform operations on video data. Following prediction using a PU, a video encoder may calculate residual values for the portion of the CU corresponding to the PU. The residual values may be transformed, scanned, and quantized. A TU is not necessarily limited to the size of a PU. Thus, TUs may be larger or smaller than corresponding PUs for the same CU. In some examples, the maximum size of a TU may correspond to the size of the

Nokia Exhibit 2006, p. 120
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	corresponding CU. Karczewicz, ¶ 50: In this disclosure, the term "coding" refers to encoding or decoding. Similarly, the term "coder" generally refers to any video encoder, video decoder, or combined encoder/decoder (codec). Accordingly, the term "coder" is used herein to refer to a specialized computer device or apparatus that performs video encoding or video decoding. The CBP coding techniques of this disclosure may be applicable to encoders or decoders. The encoder uses the described VLC techniques to encode the CBP, and the decoder uses reciprocal VLC techniques to decode the CBP. Karczewicz, ¶¶ 64, 66: [0064] In this disclosure, the phrase "block" refers to any size or type of video block. For example, the phrase "block" may refer to one or more of a macroblock, LCU, CU, sub-CU, TU, or PU. Moreover, blocks may refer to blocks of video data in the pixel domain, or blocks of data in a transform domain such as a discrete cosine transform (DCT) domain, a domain similar to DCT, a wavelet domain, or the like. In addition, the phrase block may refer to the luma and chroma blocks that are generally residual blocks of data in the transform domain. [0066] After generating the predictive block, the differences between the current video block being coded and the predictive block are coded as a residual block, and prediction syntax (such as a motion vector) is used to identify the predictive block. The residual block may be transformed and quantized. Transform techniques may comprise a DCT process or conceptually similar process, integer transforms, wavelet transforms, or other types of transforms. In a DCT or DCT-like process, as an example, the transform process converts a set of pixel values (e.g., residual values) into transform coefficients, which may represent the energy of the pixel values in the frequency domain. Quantization is typically applied on the transform coefficients, and generally involves a process that limits the number of bits associated with any given transform coefficient.

Nokia Exhibit 2006, p. 121
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The disclosures of Lin are also relevant to this limitation. *See, e.g.*, Lin, ¶0004: > High-Efficiency Video Coding (HEVC) is a new international video coding standard that is being developed by the Joint Collaborative Team on Video Coding (JCT-VC). HEVC is based on the hybrid block-based motion-compensated DCT-like transform coding architecture. The basic unit for compression, termed Coding Unit (CU), is a 2N×2N square block, and each CU can be recursively split into four smaller CUs until a predefined minimum size is reached. Each CU contains one or multiple Prediction Units (PUs). The PU sizes can be 2N×2N, 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N, nR×2N, or N×N, where 2N×N, 2N×nU, 2N×nD and N×2N, nL×2N, nR×2N correspond to horizontal and vertical partition of a 2N×2N PU with symmetric or asymmetric PU size division respectively. Lin, ¶0006: > The Skip and Merge modes utilize motion inference methods (MV=MVP+MVD where MVD is zero) to obtain the motion information from spatial neighboring blocks (spatial candidates) or temporal blocks (temporal candidates) located in a co-located picture. The co-located picture is the first reference picture in list 0 or list 1, which is signaled in the slice header. Lin, ¶0025: > A method and apparatus of deriving a motion vector predictor (MVP) for a current block in an Inter, Merge, or Skip mode are disclosed. Embodiments according to the present invention determine redundant MVP candidates according to a non-MV-value based criterion. The redundant MVP candidates are then removed from the MVP candidate set. In one embodiment, MVP candidates within a same PU (Processing Unit) or same MDSR (Motion Data Storage Reduction) unit are determined to be redundant. MVP candidates, that located in

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	different PUs and the different PUs are merged, are also determined to be redundant. In another embodiment, one MVP candidate is redundant if a first MVP candidate is coded as an Inter mode according to a second MVP candidate and zero MVD (Motion Vector Difference) and without scaling. In yet another embodiment, one or more of the MVP candidates are redundant if said one or more of the MVP candidates cause the second 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU to be merged into a 2N×2N PU. Similarly, one or more of the MVP candidates are determined to be redundant if said one or more of the MVP candidates cause the fourth N×N PU to be merged into a 2N×N, N×2N, 2N×nU, 2N×nD, nL×2N, nR×2N or 2N×2N PU. In another embodiment, for the second 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N 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) 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU. Lin, ¶0044: One embodiment of the present invention identifies and removes redundant MVP candidates to avoid imitation of Merge. In HEVC, a 2N×2N PU may be partitioned into 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs according to the Rate-Distortion Optimization (RDO) process, where 2N×nU, 2N×nD, nL×2N and nR×2N are asymmetrical motion partition (AMP) PUs. If the second 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU is merged with the first PU, the two 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs will be merged back into one 2N×2N PU. Therefore, the second PU should avoid the respective 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N Merge mode so that two 2N×N 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PUs will not be merged back into a 2N×2N PU. The scenario that multiple partitioned PUs from an original PU are merged back to form the original PU is termed imitation of Merge. Consequently, the MVP candidates of a current PU that may cause the current PU to be merged with other PUs to cause imitation of Merge is considered redundant and can be removed without comparing the MV values. For example, for the second PU of 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N Merge mode, those MVPs which are located within the previous (first) PUs and will naturally make this 2N×N, 2N×nU, 2N×nD, N×2N, nL×2N or nR×2N PU merge as a 2N×2N PU can be removed

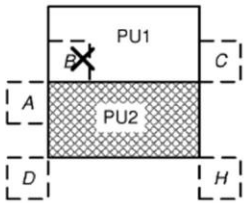
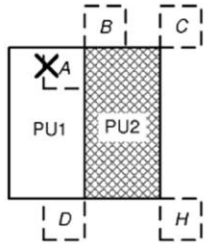
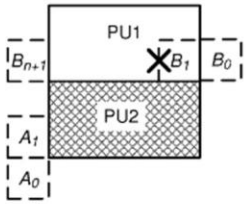
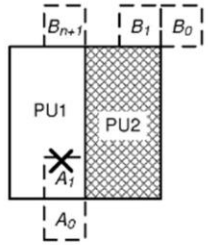
Nokia Exhibit 2006, p. 123
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	without comparing the values of MVs. FIG. 7A and FIG. 7B illustrate examples of redundancy due to imitation of 2N×N and N×2N Merge respectively. As shown in FIG. 7A, MVP B for the second 2N×N PU in the 2N×N Merge mode can be removed to avoid the duplication. In FIG. 7A, MVP A for the second N×2N PU in the N×2N Merge mode can be removed to avoid the imitation of Merge. The neighboring block configuration shown in FIGS. 7A and 7B for determining spatial/temporal MVPs are used for illustration purpose. A system may adopt other neighboring block configurations, and the MVP redundancy due to the imitation of Merge can be identified and removed according to the present invention. For example, the neighboring block configuration for the Inter and Skip modes as shown in FIG. 1 may be used for the Merge mode as well. The redundancy due to imitation of Merge can be determined as shown in FIG. 7C-FIG. 7H. In FIG. 7C, MVP B1 for 2N×N PU2 is from 2N×N PU1. The 2N×N PU1 and PU2 merging will cause the imitation of Merge. Accordingly, MVP B1 is determined to be redundant and is removed from the MVP candidate set. Similarly, MVP A1 in FIG. 7D, MVP B1 in FIG. 7E, MVP A1 in FIG. 7F, MVP B1 in FIG. 7G and MVP A1 in FIG. 7H are determined to be redundant and are removed from the MVP candidate set, respectively. Lin, ¶0047: One embodiment of the present invention identifies and removes redundant MVP candidates using motion ID. This embodiment utilizes motion ID to track the inheritance process of motion information passed from a coded PU to the current PU so as to follow the trail of the MV. The motion ID is used to help identify redundancy in MVP candidates without the need for comparing the MV values. In addition, motion ID also offers a benefit of alleviating parsing error and error propagation. In order to reduce the bits used to encode the index of motion vector predictor (MVP), redundant MVPs are removed from the MVP candidate set. However, this can cause parsing errors leading to severe error propagation. When a motion vector (MV) of a previous picture cannot be decoded correctly, a mismatch between the candidate set on the encoder side and that on the decoder side may occur, resulting in parsing error of the index of the best MVP. Consequently, the rest of the current picture cannot be parsed or decoded properly. What is even worse, this parsing error can affect subsequent inter

Nokia Exhibit 2006, p. 124
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	pictures that also use temporal MVPs. Motion identification (ID) is proposed to describe the inheritance process of motion information passed from a coded PU to the current PU to follow the trail of the MV. A unique motion ID is assigned for each new MV created in the Inter mode and can be inherited through the Merge and Skip modes. By comparing the motion ID, the MVP having the same motion ID as a previous MVP in the candidate set can thus be removed. Even if the MV of a previous picture had not been correctly decoded, the decision of redundancy will not be affected because the inherent of motion ID does not rely on the motion vector values. Lin, Fig.7A-7D:

Nokia Exhibit 2006, p. 125
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	 Fig. 7A  Fig. 7B  Fig. 7C  Fig. 7D *See also* Lin Provisional at 7, 8-11, 17 (corroborating Lin's disclosures) *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.1.2] the received block of pixels is horizontally divided into a first prediction unit and a	Rusert, as evidenced by the example citations below, discloses 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

Nokia Exhibit 2006, p. 126
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	similar motion information than a spatial motion vector prediction candidate above the prediction unit. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071:

Nokia Exhibit 2006, p. 127
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre- defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. Karczewicz, ¶¶33-36, 50, 64, 66, reproduced *supra*.

Nokia Exhibit 2006, p. 128
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Lin, ¶¶0004, 0006, 0025, 0044, 0047, Fig. 7A-7D, reproduced *supra*. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.2] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than

Nokia Exhibit 2006, p. 129
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to

Nokia Exhibit 2006, p. 130
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.2.1] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met:

Nokia Exhibit 2006, p. 131
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme.

Nokia Exhibit 2006, p. 132
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0087: > The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. Karczewicz, ¶¶33-36, 50, 64, 66, reproduced *supra*. Lin, ¶¶0004, 0006, 0025, 0044, 0047, Fig. 7A-7D, reproduced *supra*. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.2.2] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: > The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to

Nokia Exhibit 2006, p. 133
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072:

Nokia Exhibit 2006, p. 134
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre- defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.3] 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	Rusert, as evidenced by the example citations below, discloses, 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.

Nokia Exhibit 2006, p. 135
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	*See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the

Nokia Exhibit 2006, p. 136
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references*

Nokia Exhibit 2006, p. 137
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	identified in Exhibit D-10.
6[c.4] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, Rusert, ¶¶0002-5: [0002] Recent video coding standards are based on the hybrid coding principle, which comprises motion compensated temporal prediction of video frames and coding of frame residual signals. For efficient motion compensated temporal prediction, block-based motion models are used to describe the motion of pixel blocks across frames. Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames). These motion vectors are coded in the video bit stream along with the frame residual signals. [0003] At high compression ratios (or equivalently, low video bitrates), motion vector coding takes a large part of the total amount of bits, especially in recent video coding standards such as H.264/AVC where small motion compensation block sizes are used. Typically, lossless predictive coding of motion vectors is used, i.e. coding of a motion vector MV consists of first building a motion vector predictor PMV for the vector to be coded and then transmitting the difference: DMV, where DMV=MV−PMV between the motion vector and the motion vector predictor. [0004] In H.264/AVG, a PMV is derived as the median of the motion vectors of three spatially neighboring blocks. Other approaches consider also temporally neighboring blocks

Nokia Exhibit 2006, p. 138
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	(i.e. co-located in neighboring frames) for motion vector prediction. Instead of using a fixed rule for building PMV, recently approaches have been presented that explicitly signal a PMV to be used out of a set of PMV candidates, PMV_CANDS. Although this requires additional bits to signal one candidate out of the set, it can overall save bits for motion vector coding, since DMV coding can be more efficient. That is, identifying a PMV and signaling DMV can take fewer bits than independently signaling the MV. [0005] The efficiency of motion vector coding schemes which use PMV candidate signaling depends on the suitability of the available candidates in PMV_CANDS. That is, the construction of the candidate list has a major impact on the coding performance. Existing approaches typically use motion vectors from spatially surrounding blocks or temporally neighboring blocks (co-located blocks in neighboring frames). Such construction of PMV_CANDS, i.e. considering only the few surrounding blocks as a source of motion vector predictors, can be sub-optimal. Rusert, ¶0036: The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0043: Before a motion vector associated with a current motion compensation block is processed, the PMV_CANDS list used for coding the current motion vector may be updated by using motion vectors associated with surrounding blocks.

Nokia Exhibit 2006, p. 139
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071:

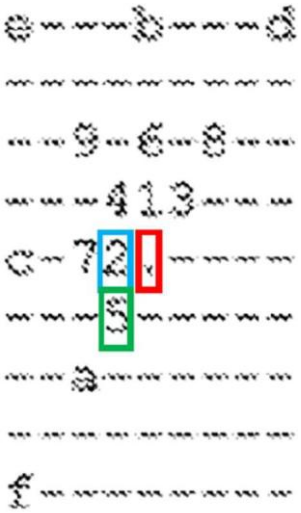
Nokia Exhibit 2006, p. 140
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0021: Any unnecessary PMV candidates may be removed from the set of PMV candidates. This ensures the length of the list is not unnecessarily long, which would reduce coding efficiency. A PMV candidate may be determined to be unnecessary if it at least one of the following conditions is fulfilled: the PMV candidate is a duplicate of another PMV candidate in the set; the PMV candidate is determined to be within a threshold distance of an existing PMV candidate; and the PMV candidate would never be used because at least one alternative PMV candidate will allow motion vectors to be coded using fewer bits. Rusert, ¶0062: A problem with this arrangement is that sometimes motion vectors close to the current block

Nokia Exhibit 2006, p. 141
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	are all exactly the same. It has been identified as advantageous to discard such duplicates. However, a problem with the scan pattern of FIG. 3 l is that a large number of the nearest neighbor backs must be investigated before any duplicates may be discarded. This means that a large number of vectors must sometimes be investigated before the candidate list can be filled and as such the above scan pattern can be considered to be slow. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. Rusert, Figs. 3a-3n, reproduced *supra*. Rusert, Fig. 3n (annotated)

Nokia Exhibit 2006, p. 142
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	 Fig. 3n : *See also* IPR2024-00604, Petition at 58-61 (explaining annotations in Fig. 3n); IPR2024-00605, Petition at 55-59 (same). The disclosures of Karczewicz are also relevant to this limitation. *See, e.g.*, Karczewicz, ¶¶ 33-36: [0033] HM refers to a block of video data as a coding unit (CU). Syntax data within a bitstream may define a largest coding unit (LCU), which is a largest coding unit in terms of the number of pixels. In general, a CU has a similar purpose to a macroblock of H.264, except that a CU does not have the same size limitations associated with macroblocks. For example, a CU may be split into sub-CUs, and the size of the LCU may be different depending on the

Nokia Exhibit 2006, p. 143
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	scenario or situation. In general, references in this disclosure to a CU may refer to a LCU of a picture or a sub-CU of an LCU. A LCU may be split into sub-CUs, and each sub-CU may be split into sub-CUs. Syntax data for a bitstream may define the size of the LCU as well as a maximum number of times an LCU may be split, referred to as maximum CU depth. Accordingly, a bitstream may also define a smallest coding unit (SCU). This disclosure also uses the term "block" to refer to any of an LCU, SCU, CU, prediction unit (PU), or transform unit (TU) (PU and TU are described in further detail below). [0034] A LCU may be associated with a quadtree data structure. In general, a quadtree data structure includes one node per CU, where a root node corresponds to the LCU. If a CU is split into four sub-CUs, the node corresponding to the CU includes four child nodes, each of which corresponds to one of the sub-CUs or one or more sub-CUs of a sub-CU. Each node of the quadtree data structure may provide syntax data for the corresponding CU. For example, a node in the quadtree may include a split flag, indicating whether the CU corresponding to the node is split into sub-CUs. Syntax elements for a CU may be defined recursively, and may depend on whether the CU is split into sub-CUs. Additional syntax elements may also be included for each CU, such as flags indicating whether filtering should be applied to the CU. [0035] A CU that is not split may include one or more prediction units (PUs). In general, a PU represents all or a portion of the corresponding CU, and includes data for retrieving a reference sample for the PU. For example, when the PU is intra-mode encoded, the PU may include data describing an intra-prediction mode for the PU. As another example, when the PU is inter-mode encoded, the PU may include data defining a motion vector for the PU. The data defining the motion vector may describe, for example, a horizontal component of the motion vector, a vertical component of the motion vector, a resolution for the motion vector (e.g., one-quarter pixel precision or one-eighth pixel precision), a reference frame to which the motion vector points, and/or a reference list (e.g., list 0 or list 1) for the motion vector. Data for the CU defining the PU(s) may also describe, for example, partitioning of the CU into one or more PUs. Partitioning modes may differ between whether the CU is intra- prediction mode encoded, or inter-prediction mode encoded. [0036] A CU may also include one or more transform units (TUs) that may be used to

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	perform transform operations on video data. Following prediction using a PU, a video encoder may calculate residual values for the portion of the CU corresponding to the PU. The residual values may be transformed, scanned, and quantized. A TU is not necessarily limited to the size of a PU. Thus, TUs may be larger or smaller than corresponding PUs for the same CU. In some examples, the maximum size of a TU may correspond to the size of the corresponding CU. Karczewicz, ¶¶ 64, 66: [0064] In this disclosure, the phrase "block" refers to any size or type of video block. For example, the phrase "block" may refer to one or more of a macroblock, LCU, CU, sub-CU, TU, or PU. Moreover, blocks may refer to blocks of video data in the pixel domain, or blocks of data in a transform domain such as a discrete cosine transform (DCT) domain, a domain similar to DCT, a wavelet domain, or the like. In addition, the phrase block may refer to the luma and chroma blocks that are generally residual blocks of data in the transform domain. [0066] After generating the predictive block, the differences between the current video block being coded and the predictive block are coded as a residual block, and prediction syntax (such as a motion vector) is used to identify the predictive block. The residual block may be transformed and quantized. Transform techniques may comprise a DCT process or conceptually similar process, integer transforms, wavelet transforms, or other types of transforms. In a DCT or DCT-like process, as an example, the transform process converts a set of pixel values (e.g., residual values) into transform coefficients, which may represent the energy of the pixel values in the frequency domain. Quantization is typically applied on the transform coefficients, and generally involves a process that limits the number of bits associated with any given transform coefficient. Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10

Nokia Exhibit 2006, p. 145
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
6[c.5] 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:	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance.

Nokia Exhibit 2006, p. 146
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased.

Nokia Exhibit 2006, p. 147
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.
6[c.5.1] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;	Rusert, as evidenced by the example citations below, discloses all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as

Nokia Exhibit 2006, p. 148
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates,

Nokia Exhibit 2006, p. 149
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.5.2] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;	Rusert, as evidenced by the example citations below, discloses the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least one of the following conditions is met: Rusert, ¶¶0065-0066:

Nokia Exhibit 2006, p. 150
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme. Rusert, ¶0087:

Nokia Exhibit 2006, p. 151
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
6[c.5.3] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.*, the disclosures set forth in limitation 6[c.4]. Rusert, ¶0044: The PMV_CANDS list may be updated such that motion vectors associated with dose motion compensation blocks are inserted towards the beginning of the PMV_CANDS list (signaled with fewer bits), whereas motion vectors associated with more far away motion compensation blocks may be inserted towards the end of the PMV_CANDS list. Possible metrics to determine how far a motion compensation block is away from the current block include: the Euclidian distance (dx2+dy2, where dx and dy are distances in the x and y directions, respectively), the Manhattan distance (the sum of absolute values, |dx|+|dy|), or the Chebyshev distance (the maximum of the absolute values, max(|dx|, |dy|), also known as maximum metric, chessboard distance, box distance, L∞ metric (L-infinity metric), L∞ norm). To this end, an outwards going scan may be performed around the current block to obtain motion vectors to update PMV_SANDS. The scan may be terminated when at least

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	one of the following conditions is met: Rusert, ¶¶0065-0066: [0065] However, even this pattern could require too many tests before growing the scan area big enough to capture distant blocks. In another alternative embodiment, all blocks other than the corner blocks and the blocks directly above and to the left of the current block are excluded. This pattern, combined with the above filtering mechanisms provides an ordering as shown in FIG. 3 n. [0066] The search pattern shown in FIG. 3 n has been found to combine good compression efficiency (i.e., finding good motion vectors) with good computation performance. Rusert, ¶0071: One measure for reducing the number of candidates is to avoid duplicate occurrences of the same motion vector in a given PMV_CANDS list. This car 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, either removing the duplicate vector or skipping the new vector. It is preferable to skip the new vector; otherwise a subsequent duplicate from a distant block may cause a candidate high in the order of the list to be put at the end of the list. Rusert, ¶0072: Removing or skipping new motion vectors may likewise be done for 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, where similarity measures could be Euclidian distance (x0−x1)2+(y0−y1)2 or absolute distance |x0−x1|+|y0−y1|, with (x0,y0) and (x1,y1) being the pair of motion vectors under consideration. Rather than a straight distance measure, another approach is to look at the number of bits required to encode the distance between the motion vectors using a given encoding scheme.

Nokia Exhibit 2006, p. 153
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0087: The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
7. The method according to claim 1 further comprising including a temporal motion prediction candidate into the motion vector prediction list.	Rusert, as evidenced by the example citations below, discloses including a temporal motion prediction candidate into the motion vector prediction list. *See, e.g.,* Claim 1. Rusert, ¶0002-0005: [0002] Recent video coding standards are based on the hybrid coding principle, which comprises motion compensated temporal prediction of video frames and coding of frame residual signals. For efficient motion compensated temporal prediction, block-based motion models are used to describe the motion of pixel blocks across frames. Each motion compensation block is assigned one motion vector (for uni-predictive temporal prediction, such as in P frames) or two motion vectors (for bi-predictive temporal prediction, such as in B frames). These motion vectors are coded in the video bit stream along with the frame residual

Nokia Exhibit 2006, p. 154
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	signals. [0003] At high compression ratios (or equivalently, low video bitrates), motion vector coding takes a large part of the total amount of bits, especially in recent video coding standards such as H.264/AVC where small motion compensation block sizes are used. Typically, lossless predictive coding of motion vectors is used, i.e. coding of a motion vector MV consists of first building a motion vector predictor PMV for the vector to be coded and then transmitting the difference: DMV, where DMV=MV−PMV between the motion vector and the motion vector predictor. [0004] In H.264/AVG, a PMV is derived as the median of the motion vectors of three spatially neighboring blocks. Other approaches consider also temporally neighboring blocks (i.e. co-located in neighboring frames) for motion vector prediction. Instead of using a fixed rule for building PMV, recently approaches have been presented that explicitly signal a PMV to be used out of a set of PMV candidates, PMV_CANDS. Although this requires additional bits to signal one candidate out of the set, it can overall save bits for motion vector coding, since DMV coding can be more efficient. That is, identifying a PMV and signaling DMV can take fewer bits than independently signaling the MV. [0005] The efficiency of motion vector coding schemes which use PMV candidate signaling depends on the suitability of the available candidates in PMV_CANDS. That is, the construction of the candidate list has a major impact on the coding performance. Existing approaches typically use motion vectors from spatially surrounding blocks or temporally neighboring blocks (co-located blocks in neighboring frames). Such construction of PMV_CANDS, i.e. considering only the few surrounding blocks as a source of motion vector predictors, can be sub-optimal. Rusert, ¶0067: 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, e.g. a H.264/AVC-style median predictor derived based on spatially neighboring blocks.

Nokia Exhibit 2006, p. 155
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.
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.	Rusert, as evidenced by the example citations below, discloses selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the block of pixels. *See, e.g.,* Claim 1. Rusert, ¶0007: > Motion vector coding can require a significant proportion of an available bitrate in video coding. Increasing the number of PMV candidates may reduce the size of the difference value (DMV) that must be signaled but requires more signaling to identify the particular PMV. Accordingly, to improve video coding efficiency an improved method and apparatus for selecting PMV candidates is required. Rusert, ¶0013: > Identifying allowed distance values means that certain distance values may not be allowed. These not-allowed distance values result in skipped layers of blocks. The size of the PMV candidate list is limited because a very large list would require long code words to identify which PMV candidate to use Skipping layers can provide increased coding efficiency by ensuring that not only the nearest neighbor previous blocks are considered, but also previous blocks. This allows a candidate list to be produced using motion vectors from a wide range of previous blocks, but that is not excessively long. Rusert, ¶0023:

Nokia Exhibit 2006, p. 156
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The method may be for video encoding or for video decoding, wherein the current block is the block being encoded or decoded respectively. Rusert, ¶0035: The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶0036: The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code "index" 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0082-0090: [0082] Scaling of a motion vector candidate according to the frame distance of the reference frame to which the motion vector candidate is applied for prediction. For example, assume a candidate motion vector MV(T−1)=(X,Y) in PMV_CANDS that has been applied for motion compensated prediction from a reference frame representing the

Nokia Exhibit 2006, p. 157
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	video at time T−1, which is next to the current frame that is assumed to represent the video at time T. Now if this candidate is obtained from PMV_CANDS to be used for prediction of a motion vector pointing to a reference frame representing the video at time T−2 (two frames next to the current frame), then the motion vector magnitude can be scaled by a factor of 2, i.e. (2*X,2*Y). Also, if a candidate motion vector (X,Y) in the PMV_CANDS list refers to the video frame at T−2 is to be used for referencing the frame at T−1, the motion vector can be scaled to (X/2,Y/2). For both these cases we may end up duplicating a candidate motion vector in which case it can be removed. Scaling of candidate motion vectors is reasonable under the assumption of linear motion. [0083] Similarly, when obtaining a motion vector predictor MV(T−1)=(X,Y) in a B frame that represents time T, and that motion vector has been applied for motion compensated prediction from a left reference frame (time T−1), and now the predictor is to be used for prediction of a vector applied for motion compensated prediction from a right reference frame (time T+1), then the sign of the motion vector predictor is inverted, i.e. (−X,−Y). [0084] The candidate predictor list size can be varied. Limiting and/or reducing the number of candidates in PMV_CANDS can be helpful to reduce the overhead of signaling which PMV is used for motion vector prediction, since shorter lists require shorter code words. On the other hand, depending on video sequence characteristics, it may be beneficial to have a larger number of motion vector prediction candidates e.g. in order to save bits for DMV coding in case of irregular motion. The following methods can be used to adapt the size of PMV_CANDS list according to video sequence characteristics. [0085] The list size can be defined in the slice/frame/picture header or in a sequence-wide header (such as parameter set), i.e. signaled from the encoder to the decoder, and thus dynamically adapted by the encoder. [0086] Candidates that have not been used for prediction during encoding of a number of previously coded blocks (according to a pre-defined threshold) can be removed from the list, thus reducing the list size.

Nokia Exhibit 2006, p. 158
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	[0087] The list size may be adapted according to similarity of candidates in the list. For example, when updating a list with a motion vector MV, the number of candidates that are similar to MV (according to a distance measure such as Euclidian or absolute distance, with a pre-defined threshold) are counted. A high count indicates a high number of similar candidates, and since it may not be necessary to have many similar candidates, at least one may be removed and the list size reduced. A low number of similar candidates on the other hand may indicate that it may be beneficial to have an additional candidate, thus the list size may be increased. [0088] As mentioned above, the candidates in PMV_CANDS are sorted and the use of one of the candidates in PMV_CANDS is signaled such that the first candidate in the list is assigned the shortest code word among the candidates and that subsequent candidates in the list are assigned code words with non-decreasing length. Such code words can be defined e.g. according to Variable Length Coding (VLC) tables. The VLC table used can depend on the maximum number of candidates in PMV_SANDS (the list size), as e.g. dynamically adapted according to the methods above. Table 1 below presents some examples for VLC codes for different maximum list sizes. The left column shows the maximum list size, also denoted as C. In the right column, the VLC codes are shown along with indexes to address candidates in the PMV_CANDS list.

Nokia Exhibit 2006, p. 159
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

TABLE 1

Example VLC codes for different maximum list sizes.

Maximum list size C	Index: VLC code
	0: - (unambiguous, no signaling necessary)
2	0: 0
	1: 1
3	0: 1
	1: 00
	2: 01
4	0: 1
	1: 00
	2: 010
	3: 011
5	0: 1
	1: 00
	2: 010
	3: 0110
	4: 0111
6	0: 1
	1: 010
	2: 011
	3: 001
	4: 0000
	5: 0001
7	0: 1
	1: 010
	2: 011
	3: 0010
	4: 0011
	5: 0000
	6: 0001

[0089] For bi-predicted motion compensation blocks, two motion vectors are coded, and

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN

thus two PMV candidates can be necessary. In that case the index numbers for the two PMV candidates can be coded together to further reduce the number of bits required for index coding. Table 2 shows an example for joint index coding, considering that both motion vectors use the same PMV_CANDS list, and that it is likely that both motion vectors use the same predictor in the PMV_CANDS list. Here idx0 and idx1 denote the indexes for first and second predictor, respectively. VLC0(idx,C) denotes a VLC for an index "idx" according to Table 1 considering a maximum list size of C.

TABLE 2

VLC code for coding of two candidates indexes associated with bi-predicted block, C: maximum size of PMV_CANDS.

Case	VLC code
idx1 < idx0	VLC0 (idx0,C) 0 VLC0 (idx1,C-1)
idx1 = idx0	VLC0 (idx0,C) 1
idx1 > idx0	VLC0 (idx0,C) 0 VLC0 (idx1-1,C-1)

[0090] Unnecessary PMV candidates are removed from the list of PMV candidates. A mechanism for removing PMV candidates is required because it may happen that some candidates in the list will never be used, since choosing a candidate with a shorter codeword and encoding the distance will give a bit sequence that is shorter or of the same length compared for all possible motion vectors. In that case, they can be removed, thereby making the list shorter and the average bit length of each index shorter. As an alternative, it may be possible to instead insert more candidates. This way, the average bit length is kept the same, but the newly inserted candidate has a chance of being useful.

Rusert, ¶¶0088-102, and related Tables, reproduced *supra*.

Nokia Exhibit 2006, p. 161
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0107: In one embodiment of the proposed solution, we use a maximum of four candidates in the list. However, in another embodiment, we use seven, and there is in principle no limit to the maximum. If we allow for a bigger maximum, the list can grow and chances increase that a suitable vector can be found. On the other hand, the number of bits needed to specify the candidate vector also increases. On top of that we get the problem that many vectors can be represented using several candidates, which is unnecessary. This redundant representation grows the more vectors are added. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
9[preamble] A method comprising:	Rusert, as evidenced by the example citations below, discloses a method. *See, e.g.,* *See limitation 1[preamble].*
9[a] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[a].*

Nokia Exhibit 2006, p. 162
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	Rusert, ¶0023: The method may be for video encoding or for video decoding, wherein the current block is the block being encoded or decoded respectively. Rusert, ¶0029: FIGS. 2 a and 2 b show the use of a PMV candidate list during encoding and decoding respectively; Rusert, ¶0035: The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶3006: The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code “index” 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise, the

Nokia Exhibit 2006, p. 163
ASUS v. Nokia
IPR2025-01153

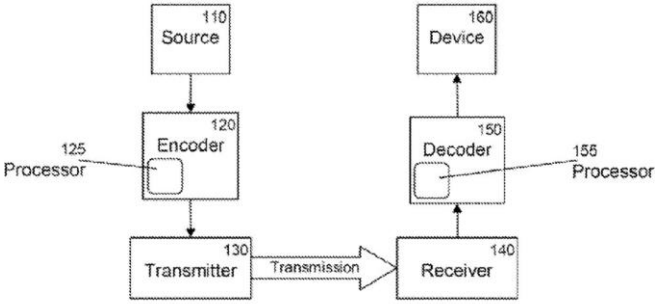
ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0037: The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
9[b] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[b].*
9[c] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[c].*

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
candidate pair from the set of spatial motion vector prediction candidates;	
9[d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and	Rusert, as evidenced by the example citations below, discloses determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing. *See, e.g.,* *See limitation 1[d].*
9[e] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[e].* Rusert, ¶0023: The method may be for video encoding or for video decoding, wherein the current block is the block being encoded or decoded respectively. Rusert, ¶0029: FIGS. 2 a and 2 b show the use of a PMV candidate list during encoding and decoding respectively;

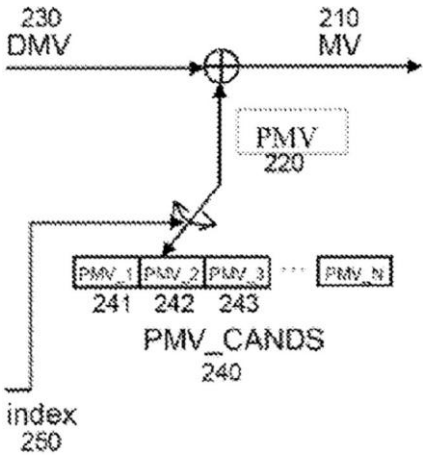
Nokia Exhibit 2006, p. 165
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0034: FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. Rusert, ¶0035: The methods disclosed herein are performed in the encoder during encoding, and also in the decoder during decoding. This is achieved even though the generation of the signaling bits is done in the encoder. During decoding the decoder parses the bits and mimics the encoder in order to achieve encoder/decoder synchronization. Because the encoder and decoder follow the same rules for creating and modifying the set of PMV candidates, the respective lists of PMV candidates stored in the encoder and decoder maintain synchronization Still, explicit signaling of PMV candidate lists may be performed under certain circumstances. Rusert, ¶0036: The described methods assume coding of a motion vector (MV) 210 using predictive coding techniques, where a predicted motion vector (PMV) 220 is used to predict a MV 210, and the prediction error or difference (DMV) 230 is found according to DMV=MV−PMV. DMV 230 is signaled from the encoder 120 to the decoder 150. Additionally, a code “index” 250 is sent to select a particular PMV candidate, in this case 242 from a list of PMV candidates, PMV_CANDS 240 as shown in FIG. 2 a. The index 250 may be sent once together with each transmitted motion vector MV 210, i.e. per sub-block (e.g. 8×8 pixel block). Likewise,

Nokia Exhibit 2006, p. 166
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	the index may be sent for groups of motion vectors, e.g. per macroblock (16×16 block). Rusert, ¶0037: The list of PMV candidates, PMV_CANDS (240) has N elements PMV_1 (241), PMV_2 (242), PMV_3 (243) etc. The list PMV_CANDS 240 is identically available at both the encoder 120 and the decoder 150. Using the transmitted index, the decoder 150 can determine the PMV 220 as used in the encoder as shown in FIG. 2 b, and thus may reconstruct MV=DMV+PMV. Rusert, ¶0040: Alternatively, one or more motion vector candidates that have not been encoded previously may be added into the PMV_CANDS list at the encoder, and then those motion vectors will be explicitly signaled to the decoder for use with PMV_CANDS, such that PMV_CANDS can be updated in the same way both at the encoder and the decoder. Rusert, Fig. 1:  Fig. 1 Rusert, Fig. 2b:

Nokia Exhibit 2006, p. 167
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	 Fig. 2b *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g., See claim 3.*
11. The method according to claim 9 further comprising examining whether the received	Rusert, as evidenced by the example citations below, discloses 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

Nokia Exhibit 2006, p. 168
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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.	list if the prediction unit is the second prediction unit. *See, e.g.,* *See claim 4.* Rusert, ¶0034: > [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
12[preamble] The method according to claim 9 further comprising	*See claim 9.*
12[a] determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and	Rusert, as evidenced by the example citations below, discloses determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list. *See, e.g.,* *See limitation 5[a].*

Nokia Exhibit 2006, p. 169
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
12[b] limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.	Rusert, as evidenced by the example citations below, discloses limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number. *See, e.g.,* *See limitation 5[b].*
13[preamble] The method according to claim 12 further comprising:	*See, e.g.,* *See claim 12.*
13[a] examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;	Rusert, as evidenced by the example citations below, discloses examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number. *See, e.g.,* *See limitation 6[a].*
13[b] if so, examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;	Rusert, as evidenced by the example citations below, discloses examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction. *See, e.g.,* *See limitation 6[b].*
13[c] if so, performing at least one of the following:	Rusert, as evidenced by the example citations below, discloses this limitation, including performing: 13[c.4] 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

Nokia Exhibit 2006, p. 170
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	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. *See, e.g.,* *See limitation 6[c].* Rusert, ¶0034: > [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
13[c.1] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,*

Nokia Exhibit 2006, p. 171
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
any of the following conditions are fulfilled:	*See limitation 6[c.1].*
13[c.1.1] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;	Rusert, as evidenced by the example citations below, discloses the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit. *See, e.g.,* *See limitation 6[c.1.1].* Rusert, ¶0034: > [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160.
13[c.1.2] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 6[c.1.2].* Rusert, ¶0034:

Nokia Exhibit 2006, p. 172
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
unit;	[0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160.
13[c.2] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.2.2].*
13[c.3] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.3].*

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
information than the spatial motion vector prediction candidate above the prediction unit;	
13[c.4] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.4].*
13[c.5] 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:	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 6[c.5].*

Nokia Exhibit 2006, p. 174
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
13[c.5.1] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;	Rusert, as evidenced by the example citations below, discloses all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list. *See, e.g.,* *See limitation 6[c.5.1].*
13[c.5.2] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;	Rusert, as evidenced by the example citations below, discloses the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit. See, e.g., See limitation 6[c.5.2].
13[c.5.3] 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.	Rusert, as evidenced by the example citations below, discloses 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 un. See, e.g., See limitation 6[c.5.3].
14. The method according to claim 9 further comprising	Rusert, as evidenced by the example citations below, discloses selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the

Nokia Exhibit 2006, p. 175
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the encoded block of pixels.	encoded block of pixels. See, e.g., See claim 8. Rusert, ¶0034: [0034] FIG. 1 shows a video coding system wherein a video signal from a source 110 is ultimately delivered to a device 160. The video signal from source 110 is passed through an encoder 120 containing a processor 125. The encoder 120 applies an encoding process to the video signal to create an encoded video stream. The encoded video stream is sent to a transmitter 130 where it may receive further processing, such as packetization, prior to transmission. A receiver 140 receives the transmitted encoded video stream and passes this to a decoder 150. Decoder 150 contains a processor 155, which is employed in decoding the encoded video stream. The decoder 150 outputs a decoded video stream to the device 160.
15[preamble] 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:	Rusert, as evidenced by the example citations below, discloses an apparatus comprising a processor and a memory including computer program code. *See, e.g.,* Rusert, ¶0001: The present application relates to a method of selecting PMV candidates, a video encoding apparatus, a video decoding apparatus, and a computer-readable medium. Rusert, ¶0024: There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a

Nokia Exhibit 2006, p. 176
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0025: There is further provided a video decoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0026: There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0027: An improved method and apparatus for selecting PMV candidates will now be described, by way of example only, with reference to the accompanying drawings, in which:

Nokia Exhibit 2006, p. 177
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	Rusert, ¶0114: The methods and apparatus described herein improve coding efficiency for motion vector prediction schemes that use signaling of motion vector predictor. Rusert, ¶0116: Further, while examples have been given in the context of particular coding standards, these examples are not intended to be the limit of the coding standards to which the disclosed method and apparatus may be applied. For example, while specific examples have been given in the context of H.264/AVC, the principles disclosed herein can also be applied to an MPEG2 system, other coding standard, and indeed any coding system which uses predicted motion vectors. Rusert, Claims 17-19: Claim 17: A video encoding apparatus comprising a processor arranged to: identify allowed distance values of distances between a current block and a previous block, wherein each Predicted Motion Vector candidate (PMV candidate) corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block; select a set of PMV candidates as a subset of a set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Claim 18: A video decoding apparatus comprising a processor arranged to: identify allowed distance values of distances between a current block and a previous block, wherein each Predicted Motion Vector candidate (PMV candidate) corresponds to a motion vector used for

Nokia Exhibit 2006, p. 178
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	coding of a previous block, said previous block having a distance from the current block; select a set of PMV candidates as a subset of a set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. 19. A computer-readable medium, carrying instructions, which, when executed by computer logic, causes said computer logic to carry out the method of claim 1. To the extent not expressly or inherently disclosed by Rusert, it would have been obvious to a POSITA to implement Rusert's teachings in the processor, memory, and computer code recited in this limitation because that is a conventional computing architecture for performing encoding operations. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
15[a] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[a].*

Nokia Exhibit 2006, p. 179
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	
15[b] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[b].*
15[c] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[c].*
15[d] determine to include or exclude the first spatial motion	Rusert, as evidenced by the example citations below, discloses determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on

Nokia Exhibit 2006, p. 180
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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	comparison of the motion information of the first spatial motion vector candidate with motion information of the spatial motion vector prediction candidate. *See, e.g.,* *See limitation 1[d].*
15[e] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[e].*
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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See claim 2.*
17. The apparatus according to claim 15, wherein the apparatus is further caused to compare motion information of the potential spatial motion vector	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g., See claim 3.*

Nokia Exhibit 2006, p. 181
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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.	Rusert, as evidenced by the example citations below, discloses examining 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. *See, e.g., See claim 4.*
19[preamble] The apparatus according to claim 15, wherein the apparatus is further caused to:	See, e.g., See claim 15.
19[a] determine a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and	Rusert, as evidenced by the example citations below, discloses determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list. See, e.g., See limitation 5[a].
19[b] limit the number of spatial motion vector prediction	Rusert, as evidenced by the example citations below, discloses limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum

Nokia Exhibit 2006, p. 182
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
candidates in the motion vector prediction list smaller or equal to the maximum number.	number. See, e.g., See limitation 5[b].
20[preamble] The apparatus according to claim 19 wherein the apparatus is further caused to:	
20[a] examine, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number;	Rusert, as evidenced by the example citations below, discloses examining, if the number of spatial motion vector prediction candidates in the motion vector prediction list smaller than the maximum number. *See, e.g.,* *See limitation 6[a].*
20[b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;	Rusert, as evidenced by the example citations below, discloses examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction. See, e.g., See limitation 6[b].
20[c] if so, perform at least one of the following:	Rusert, as evidenced by the example citations below, discloses this limitation, including performing: 20[c.4] 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

Nokia Exhibit 2006, p. 183
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	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. See, e.g., See limitation 6[c].
20[c.1] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.1].*
20[c.1.1] the received block of pixels is vertically divided into a first prediction unit and a second prediction unit;	Rusert, as evidenced by the example citations below, discloses that the received block of pixels is vertically divided into a first prediction unit and a second prediction unit. See, e.g., See limitation 6[c.1.1].
20[c.1.2] 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	Rusert, as evidenced by the example citations below, discloses that 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.

Nokia Exhibit 2006, p. 184
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
candidate has essentially similar motion information than a spatial motion vector prediction candidate above the prediction unit;	See, e.g., See limitation 6[c.1.2].
20[c.2] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.2].*
20[c.2.1] 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;	Rusert, as evidenced by the example citations below, discloses that 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. See, e.g., See limitation 6[c.2.1].
20[c.2.2] 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;	Rusert, as evidenced by the example citations below, discloses that 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. See, e.g., See limitation 6[c.2.2].
20[c.3] for a potential spatial motion vector prediction	Rusert, as evidenced by the example citations below, discloses, for a potential spatial motion vector prediction candidate, which is on a right side of the potential spatial motion vector prediction

Nokia Exhibit 2006, p. 185
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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;	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. See, e.g., *See limitation 6[c.3].*
20[c.4] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 6[c.4].*

Nokia Exhibit 2006, p. 186
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
prediction unit;	
20[c.5] 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:	Rusert, as evidenced by the example citations below, discloses, 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. See, e.g., See limitation 6[c.5].
20[c.5.1] all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list;	Rusert, as evidenced by the example citations below, discloses that all the other potential spatial motion vector prediction candidates have been included in the motion vector prediction list. See, e.g., See limitation 6[c.5.1].
20[c.5.2] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;	Rusert, as evidenced by the example citations below, discloses that the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit. See, e.g., See limitation 6[c.5.2].
20[c.5.3] 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.	Rusert, as evidenced by the example citations below, discloses that 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. See, e.g., *See limitation 6[c.5.3].*

Nokia Exhibit 2006, p. 187
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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.	Rusert, as evidenced by the example citations below, discloses including a temporal motion prediction candidate into the motion vector prediction list. *See, e.g., See claim 7.*
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.	Rusert, as evidenced by the example citations below, discloses selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the block of pixels. *See, e.g., See claim 8.*
23[preamble] 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:	Rusert, as evidenced by the example citations below, discloses an apparatus comprising a processor and a memory including computer program code. *See, e.g.,* Rusert, ¶0001: > The present application relates to a method of selecting PMV candidates, a video encoding apparatus, a video decoding apparatus, and a computer-readable medium. Rusert, ¶0024: > There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The

Nokia Exhibit 2006, p. 188
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0025: There is further provided a video decoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0026: There is further provided a video encoding apparatus comprising a processor. The processor is arranged to identify allowed distance values of distances between a current block and a previous block, wherein each PMV candidate corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block. The processor is further arranged to select a set of PMV candidates as a subset of the set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Rusert, ¶0027: An improved method and apparatus for selecting PMV candidates will now be described, by way of example only, with reference to the accompanying drawings, in which: Rusert, ¶0114:

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	The methods and apparatus described herein improve coding efficiency for motion vector prediction schemes that use signaling of motion vector predictor. Rusert, ¶0116: Further, while examples have been given in the context of particular coding standards, these examples are not intended to be the limit of the coding standards to which the disclosed method and apparatus may be applied. For example, while specific examples have been given in the context of H.264/AVC, the principles disclosed herein can also be applied to an MPEG2 system, other coding standard, and indeed any coding system which uses predicted motion vectors. Rusert, Claims 17-19: Claim 17: A video encoding apparatus comprising a processor arranged to: identify allowed distance values of distances between a current block and a previous block, wherein each Predicted Motion Vector candidate (PMV candidate) corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block; select a set of PMV candidates as a subset of a set of previously coded motion vectors that were used for previous blocks having an allowed distance from the current block. Claim 18: A video decoding apparatus comprising a processor arranged to: identify allowed distance values of distances between a current block and a previous block, wherein each Predicted Motion Vector candidate (PMV candidate) corresponds to a motion vector used for coding of a previous block, said previous block having a distance from the current block; select a set of PMV candidates as a subset of a set of previously coded motion vectors that

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	were used for previous blocks having an allowed distance from the current block. 19. A computer-readable medium, carrying instructions, which, when executed by computer logic, causes said computer logic to carry out the method of claim 1. To the extent not expressly or inherently disclosed by Rusert, it would have been obvious to a POSITA to implement Rusert's teachings in the processor, memory, and computer code recited in this limitation because that is a conventional computing architecture for performing encoding operations. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
23[a] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[a] and 1[a].*
23[b] determine a subset of	Rusert, as evidenced by the example citations below, discloses determining a subset of spatial motion

Nokia Exhibit 2006, p. 191
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
spatial motion vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate;	vector prediction candidates based on the location of the block associated with the first spatial motion vector prediction candidate. See, e.g., *See limitation 9[b] and 1[b].*
23[c] 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;	Rusert, as evidenced by the example citations below, discloses 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. See, e.g., *See limitation 9[c] and 1[c].*
23[d] 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	Rusert, as evidenced by the example citations below, discloses determining 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. *See, e.g.,* *See limitation 9[d] and 1[d].*

Nokia Exhibit 2006, p. 192
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
23[e] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[e] and 1[e].*
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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See claim 10 and claim 3.*
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	Rusert, as evidenced by the example citations below, discloses examining 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. *See, e.g.,*

Nokia Exhibit 2006, p. 193
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
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.	*See claim 11 and claim 4.*
26[preamble] The apparatus according to claim 23 wherein the apparatus is further caused to:	*See, e.g.,* *See claim 23.*
26[a] determine a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list; and	Rusert, as evidenced by the example citations below, discloses determining a maximum number of spatial motion vector prediction candidates to be included in the motion vector prediction list. *See, e.g.,* *See limitation 12[a] and 5[a].*
26[b] limit the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number.	Rusert, as evidenced by the example citations below, discloses limiting the number of spatial motion vector prediction candidates in the motion vector prediction list smaller or equal to the maximum number. *See, e.g.,* *See limitation 12[b] and 5[b].*
27[preamble] The apparatus according to claim 26 wherein the apparatus is further caused to:	
27[a] examine if the number of spatial motion vector prediction	Rusert, as evidenced by the example citations below, discloses examining if the number of spatial

Nokia Exhibit 2006, p. 194
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
candidates in the motion vector prediction list smaller than the maximum number;	motion vector prediction candidates in the motion vector prediction list smaller than the maximum number. *See, e.g.,* *See limitation 13[a] and 6[a].*
27[b] if so, examine whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction;	Rusert, as evidenced by the example citations below, discloses examining whether the prediction unit to which the potential spatial motion vector prediction candidate belongs is available for motion prediction. *See, e.g.,* *See limitation 13[b] and 6[b].*
27[c] if so, perform at least one of the following:	Rusert, as evidenced by the example citations below, discloses this limitation, including performing: 27[c.4] 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. *See, e.g.,* *See limitation 13[c] and 6[c].*
27[c.1] for a potential spatial motion vector prediction candidate on a left side of the	Rusert, as evidenced by the example citations below, discloses, for a potential spatial motion vector prediction candidate on a left side of the prediction unit, excluding the potential spatial motion vector

Nokia Exhibit 2006, p. 195
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
prediction unit, exclude the potential spatial motion vector prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled:	prediction candidate from the motion vector prediction list if any of the following conditions are fulfilled. *See, e.g.,* *See limitation 13[c.1] and 6[c.1].*
27[c.1.1] the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit;	Rusert, as evidenced by the example citations below, discloses that the received encoded block of pixels is vertically divided into a first prediction unit and a second prediction unit. *See, e.g.,* *See limitation 13[c.1.1] and 6[c.1.1].*
27[c.1.2] 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;	Rusert, as evidenced by the example citations below, discloses that 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. *See, e.g.,* *See limitation 13[c.1.2] and 6[c.1.2].*
27[c.2] for a potential spatial motion vector prediction candidate above the prediction unit, exclude the potential spatial motion vector prediction candidate from the motion	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,*

Nokia Exhibit 2006, p. 196
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
vector prediction list if any of the following conditions are fulfilled:	*See limitation 13[c.2] and 6[c.2].*
27[c.2.1] 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;	Rusert, as evidenced by the example citations below, discloses that 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. *See, e.g.,* *See limitation 13[c.2.1] and 6[c.2.1].*
27[c.2.2] 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;	Rusert, as evidenced by the example citations below, discloses that 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. *See, e.g.,* *See limitation 13[c.2.2] and 6[c.2.2].*
27[c.3] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 13[c.3] and 6[c.3].*

Nokia Exhibit 2006, p. 197
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
motion vector prediction candidate above the prediction unit;	
27[c.4] 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	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 13[c.4] and 6[c.4].*
27[c.5] 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:	Rusert, as evidenced by the example citations below, discloses, 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. *See, e.g.,* *See limitation 13[c.5] and 6[c.5].*
27[c.5.1] all the other potential	Rusert, as evidenced by the example citations below, discloses that all the other potential spatial

Nokia Exhibit 2006, p. 198
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
spatial motion vector prediction candidates have been included in the motion vector prediction list;	motion vector prediction candidates have been included in the motion vector prediction list. *See, e.g.,* *See limitation 13[c.5.1] and 6[c.5.1].*
27[c.5.2] the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit;	Rusert, as evidenced by the example citations below, discloses that the potential spatial motion vector prediction candidate has essentially similar motion information than the spatial motion vector prediction candidate above the prediction unit. *See, e.g.,* *See limitation 13[c.5.2] and 6[c.5.2].*
27[c.5.3] 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	Rusert, as evidenced by the example citations below, discloses that 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. *See, e.g.,* *See limitation 13[c.5.3] and 6[c.5.3].*
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.	Rusert, as evidenced by the example citations below, discloses selecting one motion vector prediction candidate from the motion vector prediction list to represent a motion vector prediction for the received encoded block of pixels. *See, e.g.,* *See claim 14 and claim 8.*
29[preamble] A non- transitory computer readable medium having stored thereon a computer executable program	Rusert, as evidenced by the example citations below, discloses a non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder.

Nokia Exhibit 2006, p. 199
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
code for use by an encoder, said program codes comprising instructions for:	*See, e.g.,* *See limitation 15[preamble].* To the extent not expressly or inherently disclosed by Rusert, it would have been obvious to a POSITA to implement Rusert's teachings in a non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder because that is a conventional computing architecture for performing encoding operations. *Further, to the extent Complainant contends this element is not expressly or inherently disclosed by Rusert, it would have been obvious in view of the disclosures in Rusert, in combination with the knowledge of a person of ordinary skill in the art, or in light of one or more of any of the references identified in Exhibit D-10.*
29[a] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[a].*

Nokia Exhibit 2006, p. 200
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
motion vector prediction list is signaled as the motion information for the prediction unit;	
29[b] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[b].*
29[c] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[c] and 15[c].*
29[d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and	Rusert, as evidenced by the example citations below, discloses determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing. *See, e.g.,*

Nokia Exhibit 2006, p. 201
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	See limitation 1[d].
29[e] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 1[e].*
30[preamble] 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:	Rusert, as evidenced by the example citations below, discloses a non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder. *See, e.g.,* *See limitation 9[preamble] and limitation 23[preamble].* To the extent not expressly or inherently disclosed by Rusert, it would have been obvious to a POSITA to implement Rusert's teachings in a non-transitory computer readable medium having stored thereon a computer executable program code for use by an encoder because that is a conventional computing architecture for performing encoding operations.
30[a] 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	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[a] and 1[a].*

Nokia Exhibit 2006, p. 202
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
block of pixels, where the motion vector prediction list comprises motion information of the spatial motion vector prediction candidates;	
30[b] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[b] and 1[b].*
30[c] 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;	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[c] and 1[c].*
30[d] determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing; and	Rusert, as evidenced by the example citations below, discloses determining to include or exclude the first spatial motion vector prediction candidate in the motion vector prediction list based on the comparing. *See, e.g.,*

Nokia Exhibit 2006, p. 203
ASUS v. Nokia
IPR2025-01153

ASSERTED CLAIMS	PRIOR ART REFERENCE: RUSERT, KARCZEWICZ, LIN
	See limitation 9[d] and 1[d].
30[e] 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.	Rusert, as evidenced by the example citations below, discloses 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. *See, e.g.,* *See limitation 9[e] and 1[e].*

Nokia Exhibit 2006, p. 204
ASUS v. Nokia
IPR2025-01153