



(51) International Patent Classification:
H04N 7/32 (2006.01)

(21) International Application Number:
PCT/JP2011/073149

(22) International Filing Date:
30 September 2011 (30.09.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/896,800 1 October 2010 (01.10.2010) US

(71) Applicant (for all designated States except US): **SHARP KABUSHIKI KAISHA** [JP/JP]; 22-22, Nagaiko-cho, Abeno-ku, Osaka-shi, Osaka, 5458522 (JP).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **SU, Yeping. SEGALL, Christopher A.**

(74) Agent: **HARAKENZO WORLD PATENT & TRADE-MARK**; Daiwa Minamimorimachi Building, 2-6, Tenjinbashi 2-chome Kita, Kita-ku, Osaka-shi, Osaka, 5300041 (JP).

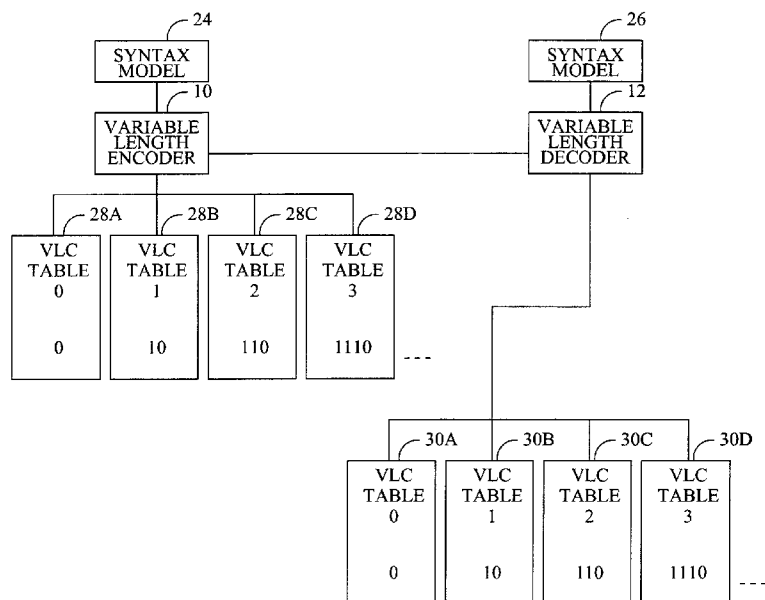
(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD,

[Continued on next page]

(54) Title: SYSTEM FOR NESTED ENTROPY ENCODING

FIG. 4



(57) Abstract: Aspects of the present invention relate to methods for decoding data. According to one aspect of the present invention, candidate motion vector is identified in a candidate set including at least one of candidate motion vectors spatially predictive, candidate motion vectors temporally predictive and candidate motion vectors spatiotemporally predictive.



RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ,
DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT,
LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS,
SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM,
GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

DESCRIPTION

TITLE OF INVENTION:

SYSTEM FOR NESTED ENTROPY ENCODING

TECHNICAL FIELD

5 Cross-Reference to Related Applications is none.

 The present invention relates to a method for trimming a candidate set of motion vectors. The present invention also relates to a system for decoding received data.

10 BACKGROUND ART

 Modern video transmission and display systems, and particularly those systems that present high-definition content, require significant data compression in order to produce a visually acceptable motion picture, because
15 transmission media simply cannot transmit an uncompressed sequence of video frames at a fast enough rate to appear as continuous motion to the human eye. At the same time, and again to produce a visually-acceptable picture, the compression technique used should not unduly sacrifice
20 image quality by discarding too much frame data.

 To achieve these dual, and conflicting goals, video compression and encoding standards such as MPEG and H.264 take advantage of temporal redundancy in the sequence

of video frames. In other words, in the vast majority of video sequences of interest to a person, adjacent frames typically show the same objects or features, which may move slightly from one frame to another due either to the movement of the object in the scene being shot (producing local motion in a frame), the movement of the camera shooting the scene (producing global motion), or both.

Video compression standards employ motion estimation to define regions in an image, which may correspond to objects, and associate with those regions a motion vector that describes the inter-frame movement of the content in each region so as to avoid redundant encoding and transmission of objects or patterns that appear in more than one sequential frame, despite appearing at slightly different locations in sequential frames. Motion vectors may be represented by a translational model or many other models that approximate the motion of a real video camera, such as rotation, translation, or zoom. Accordingly, motion estimation is the process of calculating and encoding motion vectors as a substitute for duplicating the encoding of similar information in sequential frames.

Though motion vectors may relate to the whole image, more often they relate to small regions of the image, such as rectangular blocks, arbitrary shapes, boundaries of objects, or even individual pixels. There are various methods for

finding motion vectors. One of the popular methods is block-matching, in which the current image is subdivided into rectangular blocks of pixels, such as 4x4 pixels, 4x8 pixels, 8x8 pixels, 16x16 pixels, etc., and a motion vector (or displacement vector) is estimated for each block by searching for the closest-matching block in the reference image, within a pre-defined search region of a subsequent frame.

As implied by this discussion, the use of motion vectors improves coding efficiency for any particular block of an image by permitting a block to be encoded only in terms of a motion vector pointing to a corresponding block in another frame, and a “residual” or differential between the target and reference blocks. The goal is therefore to determine a motion vector for a block in a way that minimizes the differential that needs to be encoded. Accordingly, numerous variations of block matching exist, differing in the definition of the size and placement of blocks, the method of searching, the criterion for matching blocks in the current and reference frame, and several other aspects.

With conventional motion compensation, an encoder performs motion estimation and signals the motion vectors as part of the bitstream. The bits spent on sending motion vectors can account for a significant portion of the overall bit budget, especially for low bit rate applications. Recently, motion vector competition (MVC) techniques have been

proposed to reduce the amount of motion information in the compressed bitstream. MVC improves the coding of motion vector data by differentially encoding the motion vectors themselves in terms of a motion vector predictor and a motion vector differential, where the motion vector predictor is usually selected by the encoder from a number of candidates so as to optimize rate distortion, where the candidate motion vectors consist of previously encoded motion vectors for either adjacent blocks in the same frame and/or a subset of motion vectors in a preceding frame. In other words, just as the use of a motion vector and a differential improves coding efficiency of block data by eliminating redundancies between information in sequential frames, the coding of motion vectors can exploit redundancies in situations where motion vectors between sequential frames do not change drastically, by identifying an optimal predictor, from a limited set of previously-encoded candidates, so as to minimize the bit length of the differential. The predictor set usually contains both spatial motion vector neighbors and temporally co-located motion vectors, and possibly spatiotemporal vectors.

Even using motion vector competition techniques when encoding video, however, the necessary bit rate to preserve a desired quality is often too high for the transmission medium used to transmit the video to a decoder. What is needed, therefore, is an improved encoding system for video

transmission.

The foregoing and other objectives, features, and advantages of the invention will be more readily understood upon consideration of the following detailed description of the invention taken in conjunction with the accompanying drawings.

SUMMARY OF INVENTION

A preferred embodiment is a method for decoding data, said method comprising:

(a) storing a candidate set including at least one of candidate motion vectors spatially predictive, candidate motion vectors temporally predictive and candidate motion vectors spatiotemporally predictive of a selected motion vector;

(b) identifying one of the candidate motion vectors in said candidate set; and

(c) thereafter decoding by using said candidate motion vector.

Another preferred embodiment is a method for trimming a candidate set of motion vectors used to at least one of encode and decode a selected one of said candidate set, said method comprising:

(a) identifying a subset of motion vectors in said candidate set comprising duplicates of a motion vector

outside of said subset;

(b) removing at least one motion vector from said subset; and

(c) thereafter trimming said motion vectors in said subset from said candidate set.

Another preferred embodiment is a decoding system for decoding received data, said system comprising:

(a) a syntax modeler for associating a sequence of first symbols with selective ones of a plurality of syntax elements in said data, where said syntax elements are used to index a plurality of VLC tables;

(b) a decoder that iteratively chooses different combinations of available said syntax elements.

BRIEF DESCRIPTION OF DRAWINGS

FIGS. 1A and 1B generally illustrate motion vector competition.

FIG. 2 shows an exemplary system for encoding and decoding motion vectors.

FIG. 3 shows a nested entropy encoding structure.

FIG. 4 shows a system using the nested entropy encoding structure depicted in FIG. 3.

FIG. 5A shows an exemplary encoder capable of trimming a candidate set of motion vectors.

FIG. 5B shows an exemplary method of trimming a

candidate set of motion vectors used by the encoder of FIG. 5.

FIG. 6 generally illustrates an alternate embodiment of encoding a temporally co-located motion vector in a candidate set of motion vectors.

5

DESCRIPTION OF EMBODIMENTS

Referring to FIGS. 1A and 1B, a motion vector for a candidate block (shown in cross-hatch) in a current frame at $T=0$ points to the cross-hatched block in subsequent frame at $t=1$. This motion vector may be encoded with reference to a candidate set of motion vectors V_a , V_x , V_y , and V_z . In this example, motion vector V_a is a co-located motion vector in the preceding frame at $t = -1$ and points to block A in the current frame. Motion vectors V_x , V_y , and V_z are previously-encoded motion vectors in the current frame and point to blocks X, Y, and Z, respectively, in the subsequent frame at $T=1$. FIG. 1A also shows the blocks A', X', Y', and Z' that the respective motion vectors would point to if used when encoding the candidate block.

As can be seen in FIG. 1B, , using the motion vector competition (MVC) procedure, motion vector V_z would be selected to minimize the code length of the differential V_d , which in that instance, would only require a value of "1" in a single component (down) of the vector. All other differential motion vectors either would require encoding two components

25

or would have a larger value for a single component.

It should be understood that the foregoing illustration was simplified in that different block sizes may be used, each block may represent a single pixel, and many more motion
5 vectors could be included in the candidate set. For example, all motion vectors previously calculated in the current frame could be included in the candidate set, as well as any motion vectors calculated for preceding frames. Moreover, the candidate set may include a desired number of arbitrary
10 motion vectors useful to capture large and sudden motions in the scene.

Referring to FIG. 2, and continuing with the preceding example, the selected motion vector V_z will need to be encoded. One straightforward approach is for an encoder
15 to assign a value to each candidate motion vector in a table of symbols, which assuming a variable-length entropy encoding method such as Huffman or arithmetic encoding, might look something like:

	Motion Vector Candidate	Symbol
20	V_a	0
	V_x	10
	V_y	110
	V_z	1110

Note that none of the symbols are a prefix of another symbol,
25 so that the decoder 12 can correctly parse the received

bitstream by, in this example, stopping at a received zero and decode the received bitstream with reference to a corresponding table 16. Moreover, the encoder and decoder will preferably collect statistics as the bitstream is encoded and decoded and rearrange the assignments of symbols to the motion vector candidates, in the respective tables 14 and 16, so that at any given time the motion vector having the highest frequency receives the shortest symbol, etc. This process is generally referred to as entropy coding, and will usually result in significant, lossless compression of the bitstream. The encoder 10 and the decoder 12 use the same methodology to construct and update the tables 14 and 16 initialized from the beginning of the bitstream, respectively, so that for every symbol, the table 16 used to encode that symbol is identical to the table used to decode the symbol.

Even with entropy coding, the system shown in FIG. 2 can result in significant overhead when signaling which predictor is chosen from the set of candidate motion vectors. This is particularly true if the number of predictors is large. However, the more predictors used, the more efficiency is gained when encoding the differential motion vector. In order to further reduce the overhead of signaling which predictor is chosen, additional techniques may be employed.

First, the set of candidate motion vector predictors may be trimmed to eliminate duplicate vectors. Here, two motion

vectors are duplicate vectors when the two vectors have the same horizontal value, vertical value and reference index. The term duplicate vector is equivalent to duplicate motion vector, identical vector or identical motion vector. For example, in FIG. 1A, the vectors V_x , V_y are identical, hence one of the motion vectors can be trimmed, and as a result, the largest symbol 1110 in the table above can be eliminated. Second, knowing the size of the trimmed motion predictor set means that the last bit of the last symbol in the trimmed set can be omitted, e.g. in the previous example where one of V_x , V_y was trimmed, leaving 110 as the last symbol, this symbol may simply be encoded as 11 give that this bit sequence distinguishes over all the previous symbols in the table, and the decoder knows from the size of the trimmed set that there are no further symbols.

These two additional techniques may significantly reduce the overhead of signaling the selected motion vector predictor. However, the consequence of these techniques is that the entropy decoding of the motion vector predictor will depend on the motion predictor set. That is, a bitstream cannot be correctly parsed before the complete set of motion predictors are available and correctly constructed. Such a constraint has severe impact on the decoder's error resilience, resulting in two types of disadvantages. First is temporal dependency; if a picture is corrupted or lost, decoding of subsequent pictures

could fail in the parsing stage. Second is spatial dependency; if certain area of a picture is corrupted, decoding of subsequent areas in the same picture could fail in the parsing stage.

5 This may be a significant disadvantage. If motion vector data from either a prior frame or a current frame is lost, but needed to reconstruct the full candidate set of motion vectors, then the decoder will be unable to even parse the bitstream until an independently-coded frame is reached. This is a more
10 severe consequence than the mere inability to decode correctly parsed data due to the loss of information used to code motion vectors, differential motion vectors, and residuals, because in this latter circumstance any parsed data, subsequently received in the bitstream and that does not rely
15 on the missing data, can be decoded. Once the decoder cannot parse the bitstream, however, it has no way of decoding any subsequent symbols.

 Though counterintuitive, the tradeoff between error resilience and overhead reduction is not intractable. The
20 present inventors further realized that, just as coding efficiency gains are realized by signaling a selected one from a candidate set of motion vectors, coding efficiency gains could theoretically be achieved by signaling a selected one of a group of ordered candidate sets. This gain in coding efficiency
25 could work, not only in tandem with techniques such as

motion vector trimming and using truncated unary codes, but actually as a substitute for those techniques, i.e. preserving spatial and temporal independence when parsing the bitstream by not trimming duplicate candidate motion vectors and not truncating the highest-bit-length symbol.

Specifically, referring to FIG. 3, an encoder or a decoder may utilize a nested entropy encoding structure where one of a plurality of coded symbols 18 is assigned to each of a plurality of entropy-coded candidate set of motion vectors, shown as separate VLC tables 20. It should be understood that any particular one of the VLC tables 20 may include a motion vector set that differs from that another VLC table 20, meaning that a particular motion vector that appears in one VLC table 20 does not need to appear in all VLC tables 20. The encoder may signal one of the symbols 18 that corresponds to that one of the VLC tables 20 (candidate sets) for which the signaled motion vector has the highest frequency and therefore the smallest code length. Coded symbols 18 identifying a respective candidate set can themselves be entropy-coded, if desired, or may alternatively be encoded with a fixed length code, or any other appropriate coding technique.

Implicit in the foregoing discussion is the assumption that there is some non-random distribution among the plurality of all possible candidate sets of motion vectors. If,

for example, the respective individual candidate sets simply comprise all permutations of the symbols included in each, randomly distributed with respect to each other, there would be no reason to expect a net gain in coding efficiency because the number of candidate sets of motion vectors, needed to guarantee that a sufficient number of candidate motion vectors appear in a candidate set high enough in the table to benefit from a reduced code length, would be too large. Essentially, what efficiency gained in coding the selected one of the candidate motion vector is lost in the overhead of coding the symbol associated with the particular candidate set. This makes sense; just as the entropy coding of motion vectors works due to the predictable spatial and temporal relationship between the motion vectors, making some candidate motion vectors more likely than others, the disclosed nested entropy encoding structure would be expected to further compress the bitstream only if some of the possible permutations of symbols in the candidate set are more likely than others, such that the higher-code-length candidate sets are not used as often as the lower-code-length candidate sets.

Upon investigation, the present inventors discovered that, not only does the disclosed nested entropy encoding structure in fact improve coding efficiency, but the syntax elements of neighboring pixels or blocks of pixels are

correlated with the probabilities of the ordering of candidate motion vectors in a set. Referring to FIG. 4, for example, an encoder 10 may have access to syntax symbols from a syntax model 24 that defines a set of syntax elements in the encoded data to be used to differentiate multiple VLC tables of candidate sets of motion vectors, and therefore also defines a set of syntax elements used by the encoder and decoder to determine the VLC table with which to encode the selected ones of the candidate motion vectors with code symbols.

These syntax elements could for example, relate to selected candidate motion vectors in spatially or temporally neighboring blocks of pixels, relate to combinations of such selected candidate motion vectors, or alternatively relate to any factor determined to have a relationship to the probability distribution of selected motion vectors in a candidate set. In one embodiment, an encoder 10 (and hence a decoder 12) will include a learning agent that tries different combinations of syntax elements so as to intelligently maximize coding efficiency. Stated differently, the encoder 10 intelligently optimizes coding efficiency by iteratively choosing different combinations of available said syntax elements, measuring a change in coding efficiency following each chosen combination, and responding accordingly by replacing one or more syntax elements in the combination.

With the syntax symbol from the syntax model 24, the

- 15 -

encoder 10 may then use an applicable motion vector symbol for the selected motion vector for the current block from a VLC table 28a, 28b, 28c, 28d, etc, and encode the motion vector symbol in a bitstream to the decoder 12. The encoder

5 10 also updates the order of the motion vector symbols in the VLC table used based on the selected symbol. In one embodiment, any change in the frequency distribution of symbols in a table results in the symbols being reordered. In an alternate embodiment, the encoder 10 (and the decoder 12)

10 keeps track of the most frequently-occurring symbol in the un-reordered set and ensures that that symbol is at the top of the table, i.e. that it has the smallest code length. Note that, in this example, because the syntax symbol is determined solely by the syntax of previously-encoded data, the encoder

15 need not encode the syntax symbol along with the motion vector symbol, so long as the decoder 12 uses the same syntax model to determine the particular VLC table 30a, 30b, 30c, and 30d, from which to extract the received motion vector symbol. In other words, when the encoder 10 uses the

20 syntax of the previously-encoded data to differentiate the VLC tables, updating the order of symbols in those tables in the process, a very high degree of coding efficiency can be achieved.

When the decoder 12 receives a coded bitstream from the

25 encoder 10, the decoder parses the bitstream to determine the

- 16 -

relevant VLC table for a received symbol, using a syntax model 26 if available, to decode the received symbols to identify the selected motion vector from the candidate set. The decoder also updates the respective VLC tables in the same manner as does the encoder 10.

The motion vector predictor set may contain candidate motion vectors spatially predictive of a selected motion vector (i.e. candidates in the same frame as the current block), candidate motion vectors temporally predictive of a selected motion vector (i.e. candidates at the co-located block in the frame preceding the current block), and candidate motion vectors spatiotemporally predictive of a selected motion vector (i.e. candidates in the frame preceding the current block spatially offset from the co-located block). As noted previously, the disclosed nested entropy encoding structure permits a decoder to parse a bitstream without trimming candidate motion vectors or truncating code symbols, thereby preserving spatial and temporal independence in the parsing process, and preserving error resilience while at the same time achieving significant coding efficiencies. Alternatively, the nested entropy encoding structure can be used in tandem with the techniques of trimming candidate motion vectors or truncating code symbols, while at least partially preserving error resilience.

For example, referring to FIG. 5A, an encoder 10 may

include a candidate motion vector set construction module 40 that retrieves from one or more buffers 28 the full set of candidate motion vectors applicable to a current block being encoded. A candidate motion vector set trimming module 42
5 then selectively trims the set of candidate motion vectors according to predefined rules, by applying a syntax model 24 to the set of candidate motion vectors, prior to encoding a selected motion vector with an encoding module 44, which in turn selects a symbol based on the trimmed set of candidates.
10 One potential predefined rule, for example, may prevent the candidate motion vector set module 42 from trimming motion vector predictors derived from previously reconstructed/transmitted frames. In other words, in the case that two motion vector predictors have the same value
15 but one motion vector predictor corresponds to data in a current frame and a second motion vector predictor corresponds to data in a second frame, the two motion vector predictors are both included in the trimmed set. This preserves temporal independence. Another potential
20 predefined rule, for example, may prevent the candidate motion vector set module 42 from trimming motion vector predictors with different reference indices. In other words, in the case that two motion vector predictors have a same horizontal value and vertical value but the reference index of
25 a first motion vector predictor is not equal to the reference

index of a second motion vector predictor, the two motion vectors predictors are both included in the trimmed set. As another potential predefined rule, for example, two motion vectors that have a same reference index values but not a same horizontal value and/or vertical value are both included in the trimmed set. As yet another potential predefined rule, for example, when two motion vectors have the same reference index value, horizontal value and vertical value are in the set, one is removed by the candidate motion vector set module 42.

As another example, a predefined rule may prevent the candidate motion vector set trimming module 42 from trimming motion vector predictors derived from regions that are located in different slices, so as to preserve spatial independence. As an additional embodiment, a predefined rule may prevent the candidate motion vector set trimming module 42 from trimming motion vector predictors derived from regions that are located in different entropy slices, where an entropy slice is a unit of the bit-stream that may be parsed without reference to other data in the current frame.

These two rules are stated for purposes of illustration only, as additional rules may be created as desired. FIG. 5B, for example, shows a generalized technique for applying any one of a wide variety of trimming rule sets that are signaled using a novel flag. At step 50, an encoder 10 receives a candidate set of motion vector predictors from a buffer 28, for

example. At step 52 a flag is signaled by the encoder (or received by the decoder) that is used at decision step 53 to indicate whether trimming is applied, and optionally a trimming rule set as well that may be used to define which vectors will be trimmed. If the flag indicates that no trimming is to occur, the technique proceeds to step 60 and encodes the selected motion vector using the full set of candidate motion vectors. If, however, the flag indicates that, under a given rule set, trimming is to occur, then the subset of duplicate motion vectors is identified in step 54. Thus, the subset of duplicate motion vectors can be considered in one embodiment as a maximized collection of motion vectors for which each member of the subset has an identical motion vector not included in the subset. In other words, the subset may be seen as one that excludes from the subset any motion vector in the full set of candidates that has no duplicate and also excludes from the subset exactly one motion vector in a collection of identical duplicates.

At step 56, according to predefined rules of the rule set, selected candidate motion vectors may be selectively removed from the subset of duplicates. It is this step that enables spatial and/or temporal independence to be preserved. Optionally, candidate motion vectors can also be added to the subset of duplicate motion vectors, for reasons explained in more detail below. Stated on a conceptual level, the purpose

of steps 54 and 56 is simply to apply a rule set to identify those motion vectors that will be trimmed from the full candidate set. Once this subset has been identified, the candidate motion vectors in this subset is trimmed at step 58 and the encoder then encodes the selected motion vector, from those remaining, based on the size of the trimmed set at step 60.

To illustrate the functionality of the generalized technique shown in FIG. 5A, consider the example of a temporal_mvp_flag used by the encoder to signal into the bitstream a true/false condition of whether the selected motion vector, from the candidate set, is a temporally-located motion vector. Also, initially assume that the applicable rule set for this flag is intended to preserve temporal independence. If the temporal_mvp_flag indicates that a temporal predictor is selected by the encoder, the temporal predictor subset in the candidate set will not be trimmed, because to do so would create temporal dependency. However, the spatial predictor subset of the candidate set can be trimmed because the decoder 12 has foreknowledge of the size of the temporal predictor subset.

If, on the other hand, the temporal_mvp_flag signals that a temporal predictor is not selected by the encoder, the candidate set can not only be trimmed of duplicates, but in some embodiments can also be trimmed of temporal

predictors, resulting in a drastically diminished candidate set that needs to be encoded. It should also be recognized that, if an applicable rule set permits both temporal and spatial dependencies, the a temporal_mvp_flag can be used, regardless of its value, to trim duplicates of the temporal or spatial subset signaled by the flag and to trim the entire subset not signaled by the flag.

As it happens, the inventors have determined that there is a reasonable correlation between the value of the disclosed temporal_mvp_flag and the value of a constrained_intra_pred_flag, associated with a frame, and often used in an encoded video bit stream. Specifically, the inventors have determined that there is a strong correlation between these two flags when the value of the constrained_intra_pred_flag is 1, and a substantially less strong correlation when the value of the constrained_intra_pred_flag is 0. Accordingly, to save overhead in signaling a selected motion vector, the encoder may optionally be configured to not encode the disclosed temporal_mvp_flag when the constrained_intra_pred_flag is set to 1 for the frame of a current pixel, such that the decoder will simply insert or assume an equal value for the temporal_mvp_flag in that instance, and to otherwise encode the temporal_mvp_flag. Alternatively, the disclosed temporal_mvp_flag may simply be assigned a value equal to

- 22 -

the constrained_intra_pred_flag, but preferably in this latter circumstance the value of a 0 should be associated in the defined rule set as causing the result of simply trimming duplicate vectors in the candidate set.

5 The disclosed nested entropy encoding structure can be additionally applied to this temporal_mvp_flag syntax. In one embodiment, top and left neighboring flags are used to determine the predictor set template used in the entropy coding of temporal_mvp_flag. This may be beneficial if, as is
10 the usual case, the encoder and decoder exclusively assigns entropy symbols to coded values, and also where the temporal_mvp_flag may take on many values. In another embodiment, the predictor set template for the coding of the selected motion vector for the candidate set is made
15 depending on the temporal_mvp_flag of the current block.

 Also, another embodiment of the invention signals if the motion vector predictor is equal to motion vectors derived from the current frame or motion vectors derived from a previously reconstructed/transmitted frame, as was
20 previously described with respect to the temporal_mvp_flag. In this particular embodiment, however, the flag is sent indexed by the number of unique motion vector predictors derived from the current frame. For example, a predictor set template in this embodiment could distinguish all possible
25 combinations of a first code value that reflects the

combination of flags in the two blocks to the left and above the current block, e.g. 00, 01, 10, 11 (entropy coded as 0, 10, 110, and 1110) as indexed by a second code value reflective of the number of unique motion vectors in the candidate set.

5 Alternatively, a context template in this embodiment could identify all possible combinations of a first code value that reflects whether the flags in the two blocks to the left and above the current block are identical or not, e.g. 00 and 11 entropy coded as 0 and 01 and 10 entropy coded as 10, for
10 example, and a second code value reflective of the number of unique motion vectors in the candidate set.

An encoding scheme may include a candidate set of motion vectors that includes a large number of temporally co-located motion vectors from each of a plurality of frames,
15 such as the one illustrated in FIG. 6. This means that, to encode the blocks 64 of a current frame, the encoder may have to access one or more buffers that contains a history of all the selected motion vectors in each of the prior frames from which a candidate motion vector is extracted. This can
20 require an extensive amount of memory. As an alternative, the smaller-sized block of pixels used in the encoding scheme, e.g. a 2x2 block, may be grouped in larger blocks 62, where the motion vectors stored in the buffer, and later used as co-located motion vectors when encoding subsequent blocks,
25 may instead be the average motion vector 66 of all the

selected vectors in the respective group. This trades memory requirements for coding efficiency, as the averaging procedure tends to produce a larger differential to be encoded whenever the co-located motion vector is selected. Having said that, the reduction in coding efficiency is not all that great given that the averaged co-located vector will only be chosen if it is more efficient to use that vector than any of the alternatives in the candidate set. In addition to using an average of adjacent blocks, a vector median operation or a component-wise medial operation may be used, as can any other standard operation such as maximum, minimum, or a combination of maximum and minimum operations, commonly called a dilate, erode, open, or close operation. Furthermore, the motion vector from a predefined location may be used. For example, the motion vector corresponding to the N -th smaller-sized block of pixels in a larger block may be stored in the buffer for latter use as co-located motion vectors for the larger block, where N is an integer corresponding to the location of the smaller-sized block in the larger block in raster scan order.

In some embodiments, the operation used to group smaller-sized blocks of pixels into larger blocks may be signaled in a bit-stream from an encoder to a decoder. For example, the operation may be signaled in a sequence parameter set, or alternatively, the operation may be signaled in the picture parameter set, slice header, or for any defined

group of pixels. Furthermore, the operation can be determined from a level or profile identifier that is signaled in the bit-stream.

In some embodiments, the number of smaller-sized blocks that are grouped to larger blocks may be signaled in a bit-stream from an encoder to a decoder. For example, said number may be signaled in the sequence parameter set, or alternatively the number may be signaled in the picture parameter set, slice header, or for any defined group of pixels. The number may be determined from a level or profile identifier that is signaled in the bit-stream. In some embodiments, the number may be expressed as a number of rows of smaller-sized blocks and a number of column of smaller-sized blocks.

It should be understood that the preceding embodiments of an encoder and/or a decoder may be used in any one of a number of hardware, firmware, or software implementations. For example, an encoder may be used in a set-top recorder, a server, desktop computer, etc., while a decoder may be implemented in a display device, a set-top cable box, a set-top recorder, a server, desktop computer, etc. These examples are illustrative and not limiting. If implemented in firmware and/or software, the various components of the disclosed encoder and decoder may access any available processing device and storage to perform the described techniques.

The terms and expressions that have been employed in the foregoing specification are used therein as terms of description and not of limitation, and there is no intention in the use of such terms and expressions of excluding
5 equivalents of the features shown and described or portions thereof, it being recognized that the scope of the invention is defined and limited only by the claims which follow.

CLAIMS

1. A method for decoding data, said method comprising:

5 (a) storing a candidate set including at least one of candidate motion vectors spatially predictive, candidate motion vectors temporally predictive and candidate motion vectors spatiotemporally predictive of a selected motion vector;

10 (b) identifying one of the candidate motion vectors in said candidate set; and

(c) thereafter decoding by using said candidate motion vector.

15 2. The method of claim 1 where said candidate set comprises a motion vector corresponding to smaller-sized adjacent blocks grouped in a larger block.

20 3. The method of claim 2 where said motion vector corresponding to adjacent blocks is selected from motion vectors of said smaller-sized adjacent blocks in said larger block.

4. The method of claim 2 where signaling whether said smaller-sized adjacent blocks are grouped in a larger block.

25 5. The method of claim 2 where said candidate motion

vectors are used as co-located motion vectors when encoding subsequent blocks.

6. The method of claim 1 where said candidate set
5 comprises candidate motion vectors that include a plurality of temporally co-located motion vectors from each of a plurality of frames.

7. The method of claim 1 where removing a duplicates
10 of a motion vector from said candidate set.

8. The method of claim 1 where rearrangement of said candidate set is associated lower bit-length symbols with said candidate set coded more frequently.

9. A method for trimming a candidate set of motion
15 vectors used to at least one of encode and decode a selected one of said candidate set, said method comprising:

(a) identifying a subset of motion vectors in said
20 candidate set comprising duplicates of a motion vector outside of said subset;

(b) removing at least one motion vector from said subset; and

(c) thereafter trimming said motion vectors in
25 said subset from said candidate set.

10. The method of claim 9 where subset is a maximized collection of motion vectors for which each member of the subset has an identical motion vector not included in the subset.

11. The method of claim 9 including the step of adding at least one motion vector to said subset, after said subset is identified and prior to trimming said motion vectors in said subset.

12. The method of claim 11 where the at least one motion vector added does not have a duplicate in said candidate set.

13. The method of claim 9 where said step of removing at least one motion vector from said subset preserves temporal independence in parsing data including said selected one.

14. The method of claim 9 where said step of removing at least one motion vector from said subset preserves spatial independence in parsing data including said selected one.

15. The method of claim 9 where the motion vector

removed from said subset was derived from a region located in different slice than a current block.

16. The method of claim 15 where said slice is an entropy slice.

17. The method of claim 9 where the motion vector removed from said subset was derived from a block in a frame different than a current block.

18. The method of claim 9 where said step of removing at least one motion vector from said subset preserves temporal and spatial independence in parsing data including said selected one.

19. A decoding system for decoding received data, said system comprising:

(a) a syntax modeler for associating a sequence of first symbols with selective ones of a plurality of syntax elements in said data, where said syntax elements are used to index a plurality of VLC tables;

(b) a decoder that iteratively chooses different combinations of available said syntax elements.

20. The decoding system of claim 19 including at least one buffer for storing a plurality of tables, each nested within

a respective one of said sequence of first symbols, and each used to associate a respective one of a sequence of second symbols with encoded data.

5 21. The decoding system of claim 19 where said step of iteratively choosing different combinations of available said syntax elements simulates an encoding process that intelligently improves coding efficiency for particular data.

10 22. The decoding system of claim 19 where said sequence of first symbols is entropy coded.

 23. The decoding system of claim 19 where said VLC
15 tables associate entropy coded symbols with candidate motion vectors.

 24. The decoding system of claim 23 where said
20 decoder uses said syntax elements to rearrange the association of said entropy coded symbols and said candidate motion vectors.

 25. The decoding system of claim 24 where the rearrangement associates lower bit-length symbols with data coded more frequently.

26. The decoding system of claim 23 where at least one of said syntax elements is a flag used to trim said candidate motion vectors.

5 27. The decoding system of claim 26 where said flag is a temporal_mvp_flag.

10 28. The decoding system of claim 27 where said flag is nested within a symbol indicating the number of unique motion vectors in said candidate set.

FIG. 1A

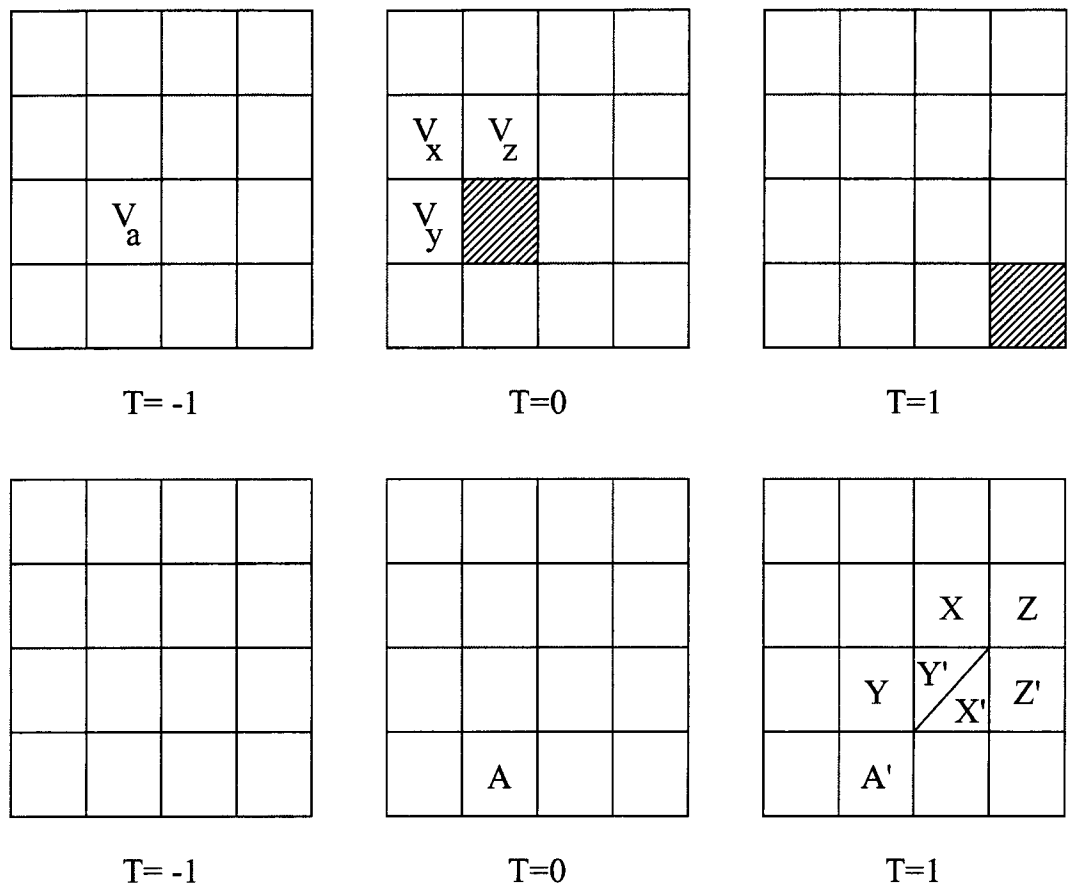
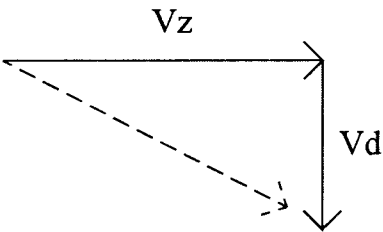


FIG. 1B



2/7

FIG. 2

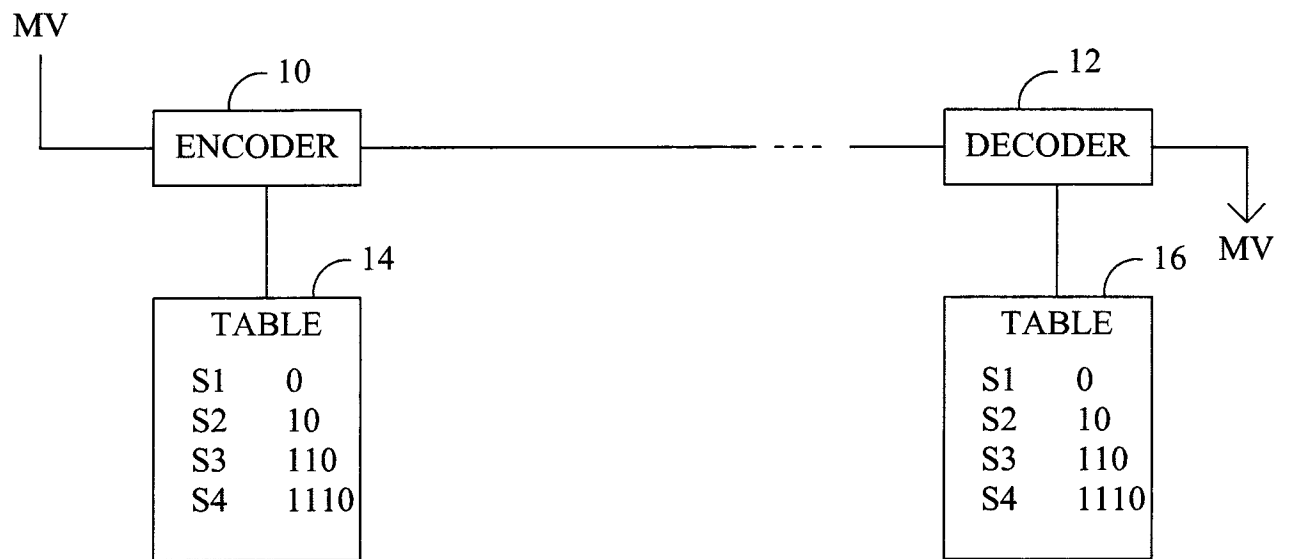
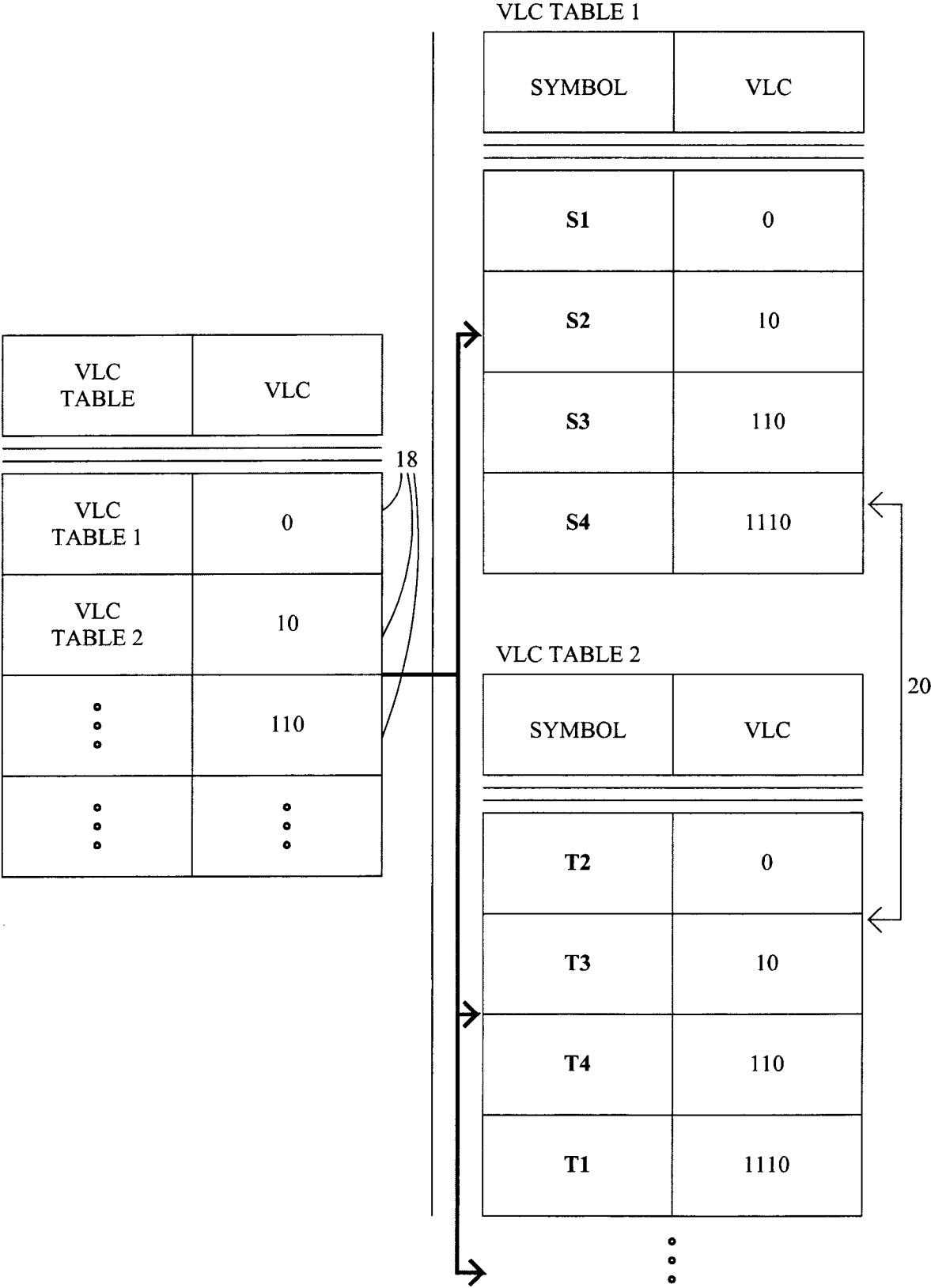
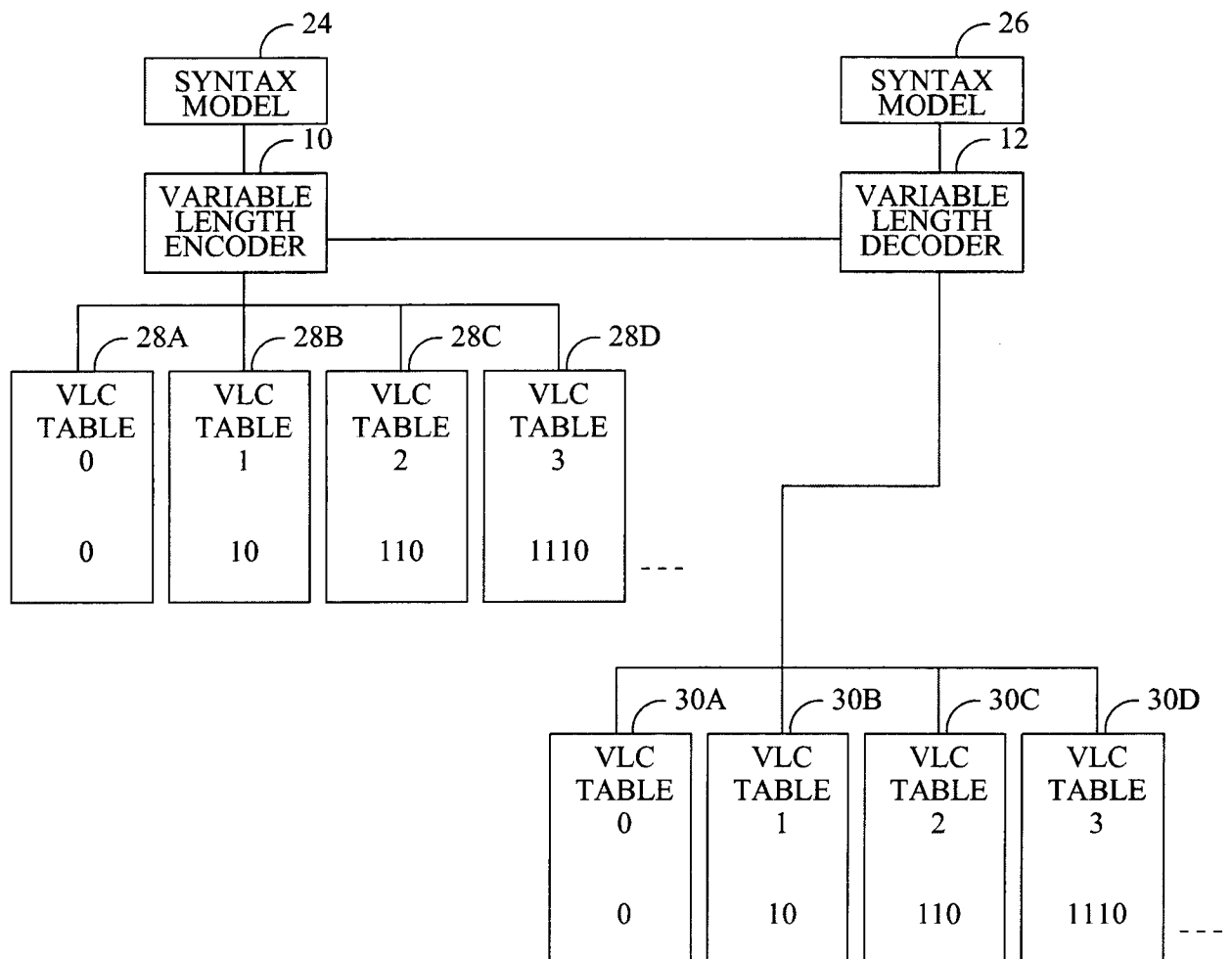


FIG. 3



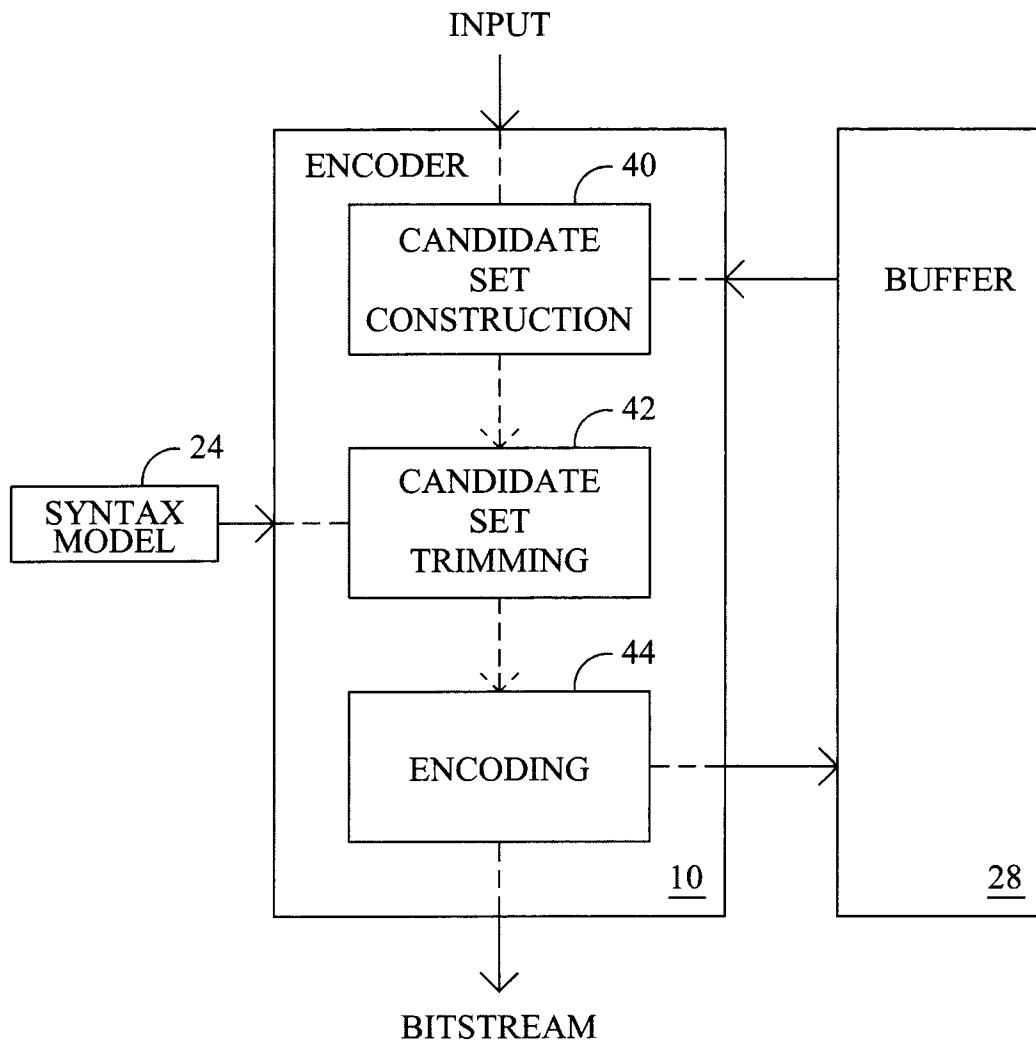
4 / 7

FIG. 4



5 / 7

FIG. 5A



6 / 7

FIG. 5B

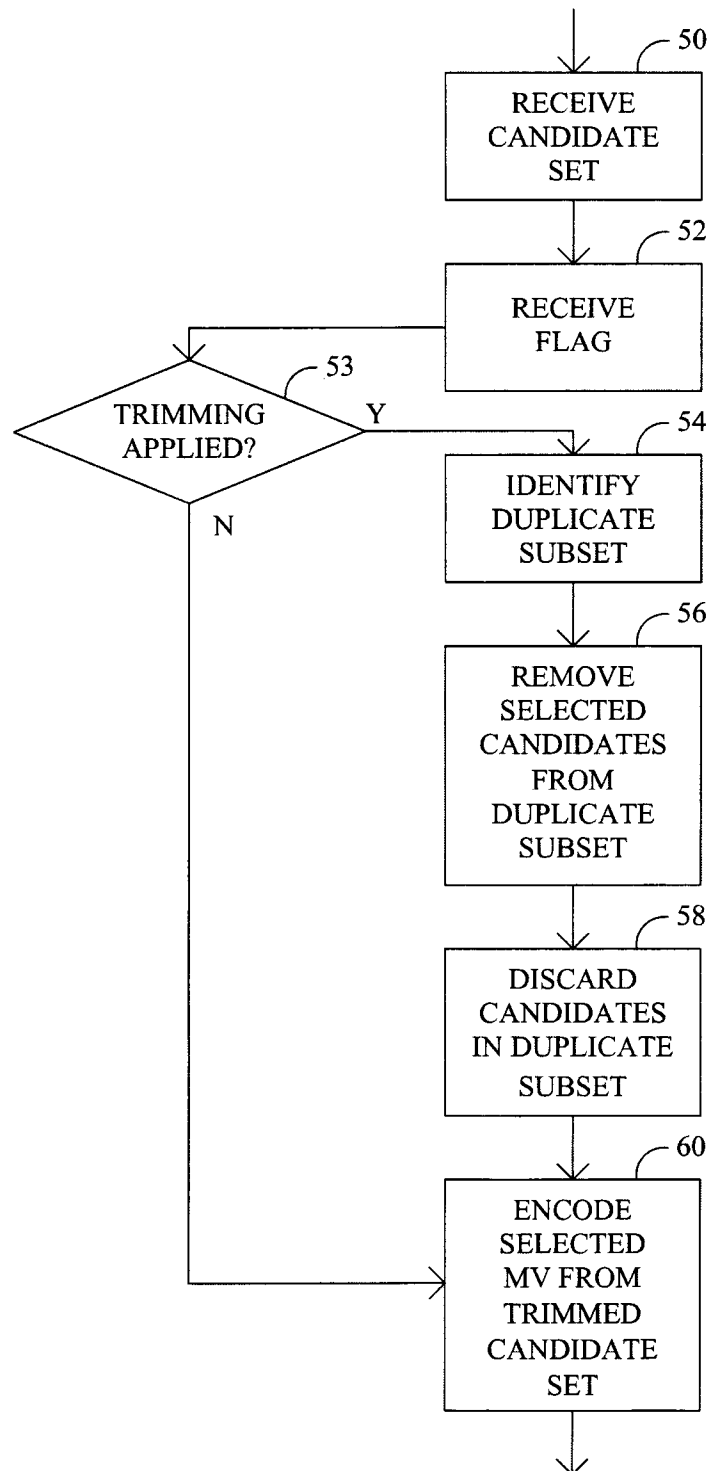
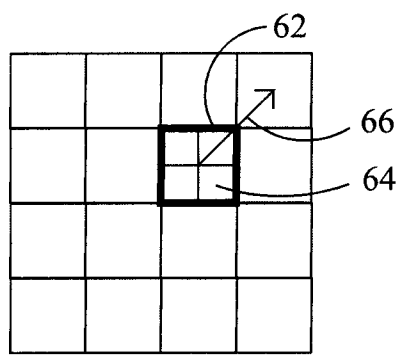


FIG. 6



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/073149

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl. H04N7/32 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl. H04N7/24-7/68

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan 1922-1996
 Published unexamined utility model applications of Japan 1971-2011
 Registered utility model specifications of Japan 1996-2011
 Published registered utility model applications of Japan 1994-2011

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y	Joel Jung, Guillaume Laroche, "Competition-Based Scheme for Motion Vector Selection and Coding", ITU - Telecommunications Standardization Sector STUDY GROUP 16 Question 6 Video Coding Experts Group (VCEG) 29th Meeting: Klagenfurt, Austria, 17-18 July, 2006, [VCEG-AC06]	1, 8 2-7
Y	JP 2007-532036 A (MICROSOFT CORPORATION) 2007.11.08, Figs.6A-9,11 & US 2005/0053292 A1 & WO 2005/027493 A2 & KR 10-2005-0025567 A & CN 1627824 A	2-5
Y	JP 10-224800 A (MATSUSHITA ELECTRIC INDUSTRIAL CO LTD) 1998.08.21, Fig.3 (Family: None)	6



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

28.12.2011

Date of mailing of the international search report

17.01.2012

Name and mailing address of the ISA/JP

Japan Patent Office

3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan

Authorized officer

Daigoro BANDO

Telephone No. +81-3-3581-1101 Ext. 3541

5C

3 2 4 1

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/073149

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	Frank Bossen, Philipp Kosse, "Simplified motion vector coding method", Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 2nd Meeting: Geneva, CH, 21-28 July, 2010, [JCTVC-B094]	7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2011/073149

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:
See an extra sheet.

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

1-8

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☐ No protest accompanied the payment of additional search fees.

Claims 1-8, claims 9-18, claims 19-28 involve neither the same nor corresponding special technical features.

Thus, there are 3 inventions in the claims of this application.

(Invention 1) claims 1-8

A method for decoding data that identifies one of the candidate motion vectors in the candidate set.

(Invention 2) claims 9-18

A method for trimming a candidate set of motion vectors.

(Invention 3) claims 19-28

A decoding system for decoding received data, comprising a syntax modeler.