Technical Disclosure Commons

Defensive Publications Series

March 11, 2015

Early Termination In Partition Search for Encoding

Yunqing Wang

Recommended Citation

Wang, Yunqing, "Early Termination In Partition Search for Encoding", Technical Disclosure Commons, (March 11, 2015)
http://www.tdcommons.org/dpubs_series/29

Early Termination In Partition Search for Encoding

ABSTRACT

An efficient partition search termination strategy suitable for high-quality encoding is proposed, which largely speeds up the encoder without sacrificing the quality. This is achieved using a partition search system. The partition search system constructs a partition search tree of an NxN encoding unit comprising multiple partition nodes to conduct a partition search. The system evaluates a partition node from the multiple partition nodes of the partition search tree. After a partition node is evaluated, the early termination checking is conducted by the system based on the information gathered during the partition evaluation. If the early termination criteria are met, the current partition size is decided as the final best choice, and all its child nodes are not searched and are removed from the search tree. Consequently, the system terminates the partition search in the current branch. However, if the early termination criteria are not met, the system goes back to evaluating the next partition node from the multiple partition nodes of the partition search tree until the early termination criteria is met or all nodes are checked.

PROBLEM STATEMENT

New-generation video codecs are capable of producing videos with 50% better quality at a given bit rate than previous-generation video codecs. Many new techniques and improvements are adopted in order to achieve the high coding efficiency. One important technique is the usage of broader range of the partition sizes, which can be from 4x4 to 64x64, e.g., 4x4, 4x8, 8x4, 8x8, 8x16, 16x8, 16x16, 16x32, 32x16, 32x32, 32x64, 64x32, and 64x64. To find the optimal partitioning for an encoding unit, an exhaustive search has to be done. Namely, the encoder goes through all possible partitionings, performs the encoding process for each partitioning to determine its rate and distortion(RD) error, and picks the partitioning that gives the least RD error. This significantly increases the computational complexity of the encoder, and consumes substantial CPU resources. An efficient partition search termination strategy suitable for high-quality encoding is proposed to speeds up the encoding process without sacrificing the quality.

PARTITION SEARCH SYSTEM

The systems and techniques described in this disclosure relate to a partition search system. The system can be implemented for use in an Internet, an intranet, or another client and server environment. The system can be implemented as program instructions locally on a client device or implemented across a client device and server environment. The client device can be any electronic device such as a mobile device, a smartphone, a tablet, a handheld electronic device, a wearable device, a laptop etc.

Fig. 1 illustrates an example method 100 for performing an efficient partition search termination strategy for use in encoding processes. Method 100 can be performed by the partition search system.

The system constructs a partition search tree of an NxN encoding unit comprising multiple partition nodes to conduct a partition search (block 110). Here, N represents an integer, such as N = 64, as in some of the new generation video codecs. The partition search tree consists of all possible partitions, for example, 4x4 to 64x64, i.e. 4x4, 4x8, 8x4, 8x8, 8x16, 16x8, 16x16, 16x32, 32x16, 32x32, 32x64, 64x32, and 64x64. The system can utilize the proposed early termination strategy with all partition types, e.g., rectangular partitions and square partitions in various sizes. As an example, an illustrative Fig. 2 shows a 4-level partition search tree including the square partitions. The early termination checking points are marked as ET in the partition search tree. The root node at level 0 (see block 210, Fig. 2) represents the encoding unit block, and the block is equally divided into 4 (N/2)x(N/2) blocks, which correspond to the 4 child nodes at level 1 (see block 220, fig. 2). This block-splitting continues until the block size equals the basic block size KxK (see block 240). The search process follows the pre-order depth-first traversal order where the parent node is evaluated before the child nodes.

After the system constructs the partition search tree, the system evaluates a partition node from the multiple partition nodes of the partition search tree (block 120). The evaluation consists of finding the partition's predictions, calculating and transforming the residuals, quantizing the transform coefficients, and de-quantizing the quantized coefficients. The best mode for the partition is the one with the least rate and distortion (RD) error. The distortion is defined as the sum of squared differences (SSD) of the transform coefficients and the de-quantized coefficients. The rate is defined as the bit rate cost for coding the quantized coefficients. The rate and distortion (RD) error is defined as the sum of the distortion and the rate multiplied by λ, where λ is the Lagrangian multiplier. A SKIP flag is defined. If all quantized coefficients are zero, the SKIP is set to 1, otherwise, it is set to 0. Further, if the partition which is evaluated has an RD error that is less than the smallest RD error from previous checked partitions, the current partition becomes the new best partition choice, and its RD error becomes the smallest RD error for the partition search. Thus, the result of the evaluation allows the system to determine the partition with the smallest RD error for partition search.

After the evaluation is done, the system conducts an early termination check by determining if early termination criteria are met (block 130). The partition with the smallest RD error is analyzed to determine if it satisfies a combination of termination criteria. The termination criteria can include SKIP is 1, rate < RATE_THRESHOLD, and distortion < DISTORTION_THRESHOLD, where DISTORTION_THRESHOLD is adjusted by number of pixels in the partition. The main factors that play an important role in partition decision making may be the local motion and the bit rates. Firstly, in a video sequence, multiple moving objects may exist and move in different directions and/or at different speeds. If a MxM partition lies on a single moving object, then the MxM partition is the best partition choice. On the other hand, if a partition lies across multiple moving objects, then it can be split into smaller partitions in order to better represent each object's motion. Secondly, a quantizer Q used to quantize the transform coefficients is selected based on the bits allocated for the current frame or the encoding unit. In high bit rate case, i.e., when Q is low, since the bits are abundant, the encoder can choose smaller partition sizes to get better predictions. In low bit rate case, i.e. Q is high, the encoder can choose larger partition sizes in order to save bits.

A strong correlation exists between the partition decisions and the SKIP values. If the quantizer Q is low, less transform coefficients are quantized to 0, less partitions have a SKIP of 1, and then less early terminations happen, which correlates with more smaller partitions chosen. If the quantizer Q is high, more transform coefficients are quantized to 0, more partitions have a SKIP of 1, and then more early terminations happen, which correlates more larger partitions chosen. Moreover, to prevent excessive early terminations, two other criteria can be included, which involve a rate threshold and a distortion threshold. Setting the distortion threshold prevents too many early terminations at low bit rates. When Q is very large, the SKIP can be 1 even if the distortion is very large, implying a poor prediction. In this case, splitting the partition and checking smaller partitions can be allowed. On the other hand, setting the rate threshold prevents too many early terminations at high bit rates. When SKIP is 1, the rate mainly consists of the costs of the mode and the motion vectors. A large rate may imply that new motion vectors are found, meaning the local motion is inconsistent. In this case, since there are enough bits, continuing to check smaller partitions for better predictions is preferred. Therefore, the early termination criteria is set accordingly.

Now, after a partition node is evaluated, the early termination checking is conducted based on the information gathered during the partition evaluation. If the early termination criteria are met, the current partition size is decided as the final best choice, and all its child nodes are not searched and are removed from the search tree. Fig. 3 shows an example of the final partitioning of the N*N encoding unit discussed in Fig. 2. Thus, if the result of the early termination check by the system is a yes, the system terminates the partition search (block 140) in the current branch without checking any child nodes below. On the other hand, if the result of the early termination check by the system is a no, the system goes back to block 120 and runs the method again until the early termination criteria are met or all nodes are checked. Additionally, because the SKIP, rate, and distortion are already calculated during the partition evaluation, no additional CPU overhead is needed.

Generally, it is relatively easy to find a good partitioning for the static or slow motion area than the fast motion area. One of the outstanding merit of this strategy is that it also works great for the moving objects such as a fast moving background.

The subject matter described in this disclosure can be implemented in software and/or hardware (for example, computers, circuits, or processors). The subject matter can be implemented on a single device or across multiple devices (for example, a client device and a server device). Devices implementing the subject matter can be connected through a wired and/or wireless network. Such devices can receive inputs from a user (for example, from a mouse, keyboard, or touchscreen) and produce an output to a user (for example, through a display). Specific examples disclosed are provided for illustrative purposes and do not limit the scope of the disclosure.

DRAWINGS

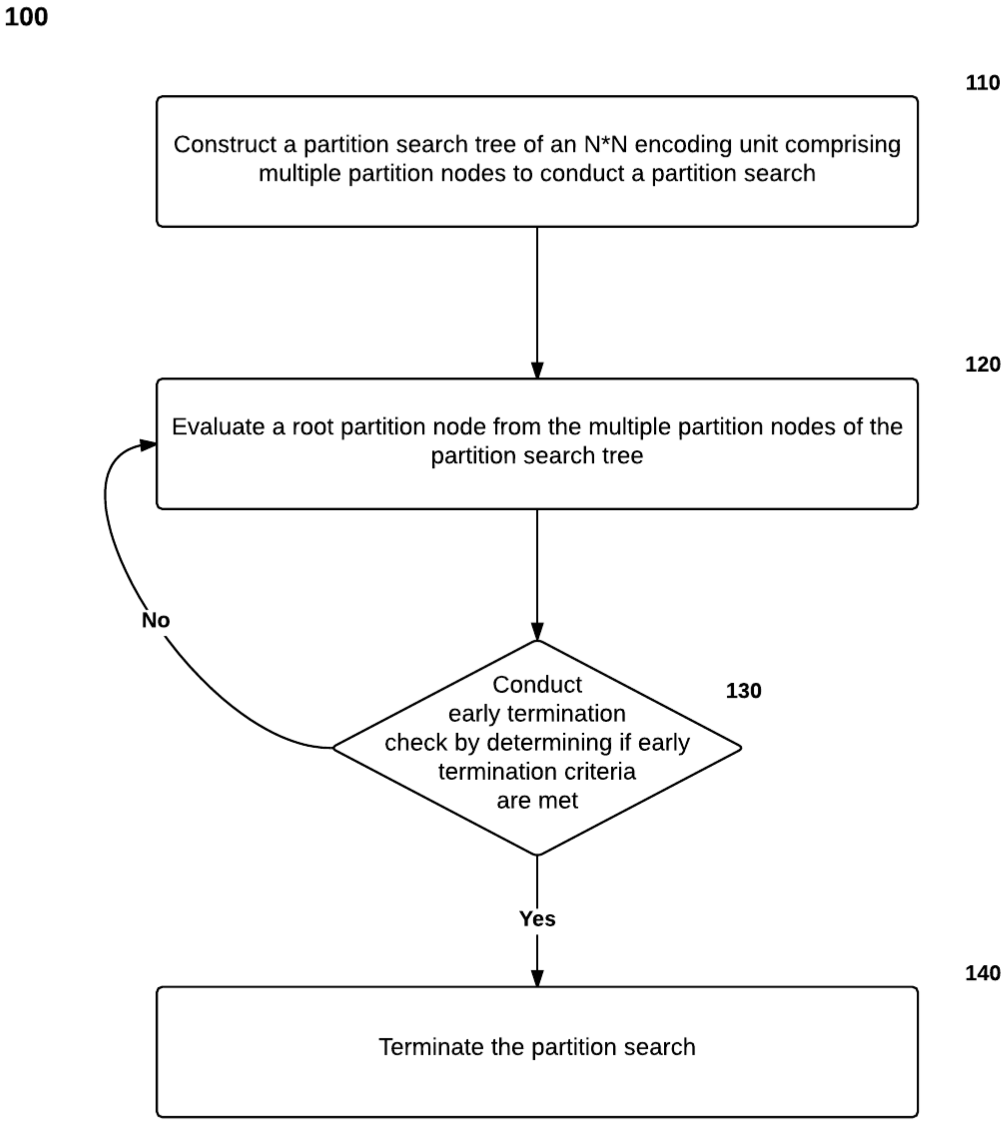
100
110
Construct a partition search tree of an N*N encoding unit comprising multiple partition nodes to conduct a partition search
120
Evaluate a root partition node from the multiple partition nodes of the partition search tree
No
Conduct early termination check by determining if early termination criteria are met
130
Yes
140
Terminate the partition search

FIG. 1

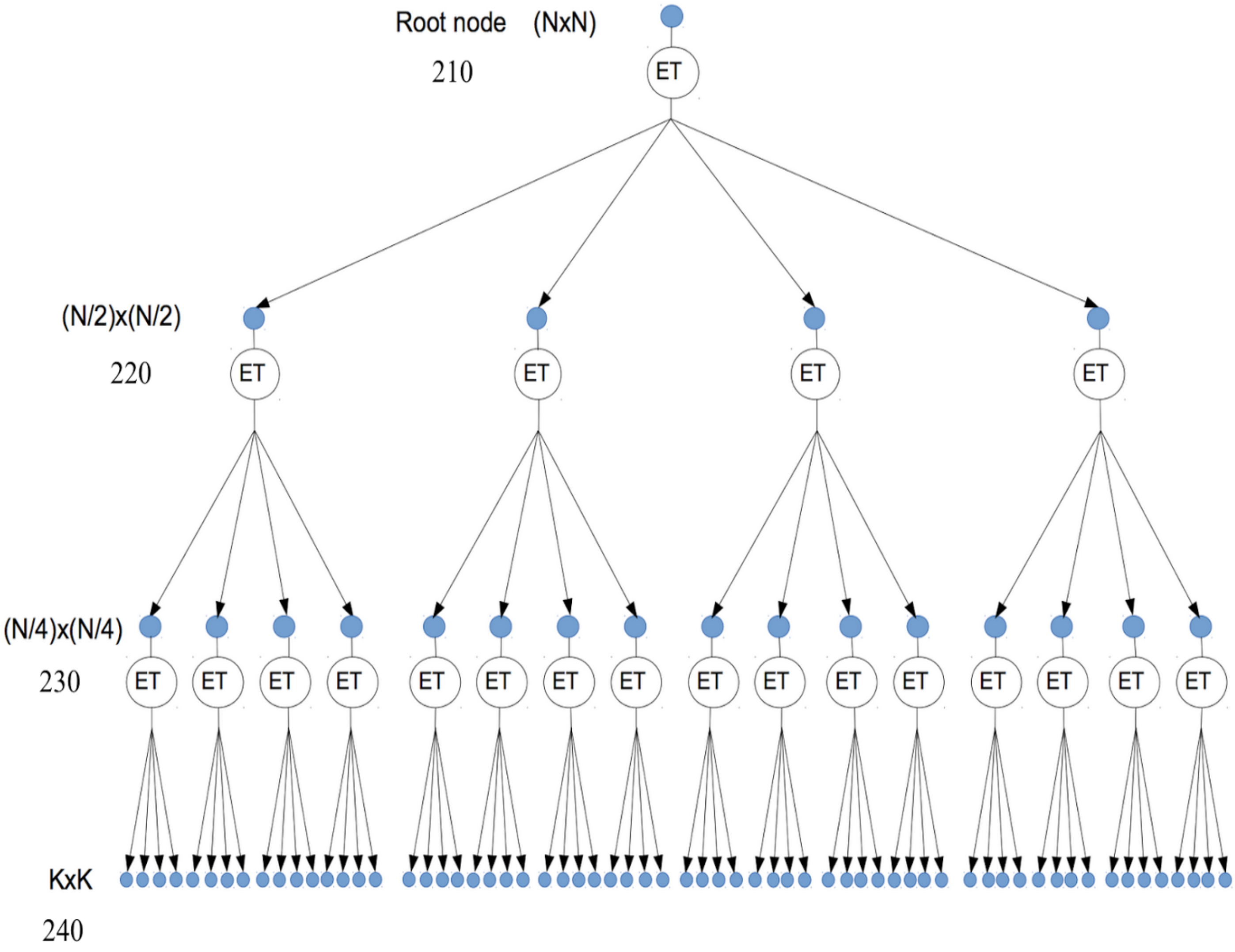

FIG. 2

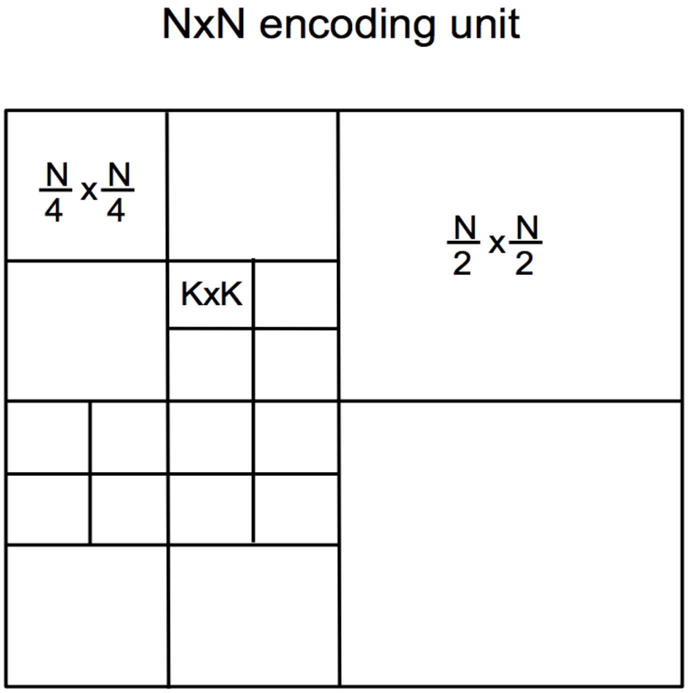
NxN encoding unit
$\frac{N}{4} \times \frac{N}{4}$
KxK
$\frac{N}{2} \times \frac{N}{2}$

FIG. 3