Technical Disclosure Commons

Defensive Publications Series

July 18, 2019

Optimal block size determination for application of a transform for file compression

Hui Su

Optimal block size determination for application of a transform for file compression

ABSTRACT

File compression often involves the use of a transform. Prior to applying a transform, the information in the file is converted to the frequency domain to make it easier to compress via the transform. Application of the chosen transform can be performed at different sizes of input data blocks. The rate-distortion (RD) cost, and correspondingly, the speed of the compression process depends on the block size chosen for application of the transform. The techniques described in this disclosure use a trained machine learning model to predict optimal block size for the application of a given transform used to compress a file. The techniques can automatically determine if it is optimal to apply the transform to the entire input block or split the block into smaller units to which the transform is subsequently applied.

KEYWORDS

- File compression
- Block size
- Rate-distortion (RD) cost
- Discrete Cosine Transform (DCT)
- Asymmetric Discrete Sine Transform (ADST)

BACKGROUND

Large files such as video, audio, media files, etc. are typically of large sizes and need to be compressed for different purposes, e.g., transfer over a network. File compression often involves the use of a transform. Prior to applying a transform, the information in the file is converted to the frequency domain to make it easier to compress via the transform. The transform applied can be one of a variety of types, such as Discrete Cosine Transform (DCT) or

Asymmetric Discrete Sine Transform (ADST). Further, application of the chosen transform can be performed at different sizes of input data blocks. The rate-distortion (RD) cost, and correspondingly, the speed of the compression process depends on the block size chosen for application of the transform.

DESCRIPTION

This disclosure describes application of a trained machine learning model (or multiple models) to predict the optimal block size for the application of a given transform used to compress a file or data. For instance, the techniques can be implemented to determine if it is optimal to apply the transform to the entire input block or split the block into smaller units to which the transform is subsequently applied.

To this end, the means and deviations of the transform input data of the entire block as well as those of half and quarter blocks of the whole block are provided as input to a trained machine learning model. Further, the corresponding means of the deviations and the deviations of the means are included as additional input to the model. Based on these inputs, the output of the trained machine learning model indicates whether the optimal block size to be used for application of the transform is the entire block or half of the block or quarter of the block. The file is then compressed by applying the transform at the indicated optimal block size.

The techniques described above can be implemented to operate recursively such that each half and/or quarter block of the original block is further split into halves and/or quarters and supplied as input to the trained machine learning model. The recursive operation can continue until the process results in the determination of an optimal block size or reaches a termination state specified based on one or more of various parameters, such as running time, memory use, efficiency, etc.

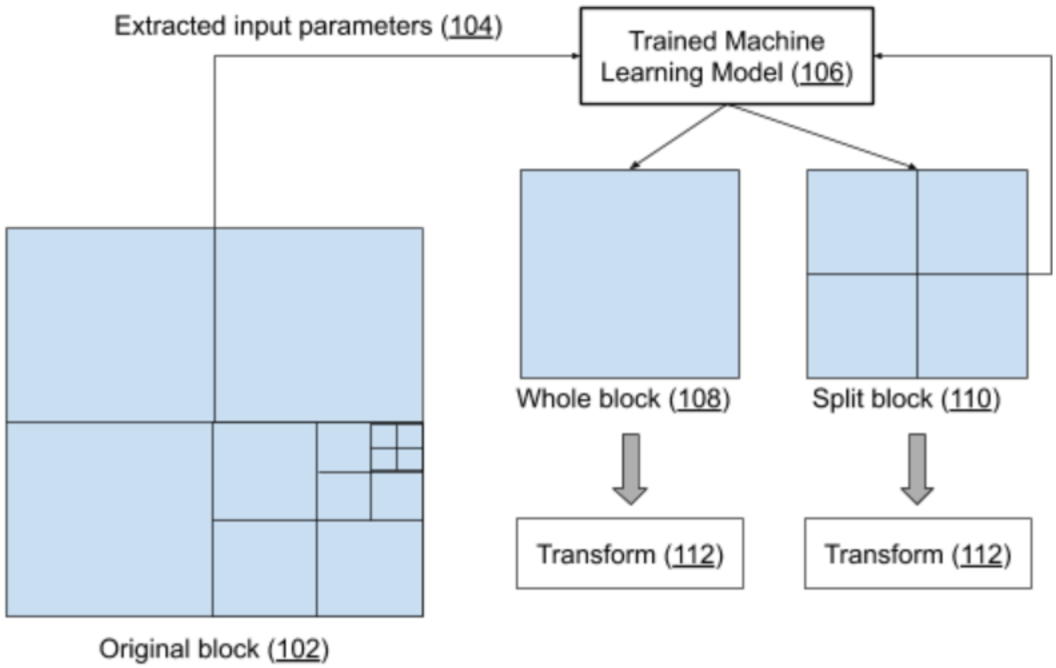

Fig. 1: Determining the optimal block size for a transform operation via machine learning

Fig. 1 shows an operational implementation of the techniques described in this disclosure. An original block of data (102) is obtained from a file that is to be compressed. As shown in Fig. 1, the block is divided in halves and quarters, each of which can be divided further into further halves and quarters. The input parameters (104) , such as means and deviations of the transform input data, are extracted from the block and supplied to a trained machine learning model (106). The output of the model indicates whether it is optimal to apply the transform to the whole block (108) or the block is to be split further (110) prior to the application of the transform (112). If appropriate, the process can be recursive such that the split block is deemed as a new original block and the relevant parameters extracted from it are fed back to the machine learning model.

The described process can use any suitable machine learning model, such as a neural network. Further, if appropriate, separate trained machine learning models can be incorporated for different block sizes. Application of the described techniques obviates the need for an exhaustive search across all possible block sizes to discover the block size that minimizes the RD cost. Such reduction in the search effort required to determine the optimal block size for applying the transform can improve the speed for the compression encoding.

The techniques can be incorporated to transform and compress files by systems, such as video or audio encoders, that use such operations. In general, the techniques can be applied within any system that requires file or data compression.

CONCLUSION

The techniques described in this disclosure use a trained machine learning model to predict optimal block size for the application of a given transform used to compress a file. The techniques can automatically determine if it is optimal to apply the transform to the entire input block or split the block into smaller units to which the transform is subsequently applied. To this end, the means and deviations of the transform input data for the entire block as well as those of half and quarter blocks of the whole block are provided as input to the trained machine learning model. Further, the corresponding means of the deviations and the deviations of the means are included as additional input to the model. Application of the described techniques obviates the need for an exhaustive search across all possible block sizes to discover the block size that minimizes the RD cost. Such reduction in the search effort required to determine the optimal block size for applying the transform can improve the speed for the compression encoding.