Input-Based Dynamic Reconfiguration of Approximate Arithmetic Units for Video Encoding

Abstract:

The field of approximate computing has received significant attention from the research community in the past few years, especially in the context of various signal processing applications. Image and video compression algorithms, such as JPEG, MPEG, and so on, are particularly attractive candidates for approximate computing, since they are tolerant of computing imprecision due to human imperceptibility, which can be exploited to realize highly power-efficient implementations of these algorithms. However, existing approximate architectures typically fix the level of hardware approximation statically and are not adaptive to input data. For example, if a fixed approximate hardware configuration is used for an MPEG encoder (i.e., a fixed level of approximation), the output quality varies greatly for different input videos. This paper addresses this issue by proposing a reconfigurable approximate architecture for MPEG encoders that optimizes power consumption with the goal of maintaining a particular Peak Signal-to-Noise Ratio (PSNR) threshold for any video. We propose two heuristics for automatically tuning the approximation degree of the RABs in these two modules during runtime based on the characteristics of each individual video. The proposed architecture of this paper analysis the logic size, area and power consumption using Xilinx 14.2.

Enhancement of the project:

Existing System:

MPEG has for long been the most preferred video compression scheme in modern video applications and devices. Using the MPEG-2/MPEG-4 standards, videos can be squeezed to very small sizes. MPEG uses both interframe and intraframe encoding for video compression. Intraframe encoding involves encoding the entire frame of data, while interframe encoding utilizes predictive and interpolative coding techniques as means of achieving compression. The interframe version exploits the high temporal redundancy between adjacent frames and only
encodes the differences in information between the frames, thus resulting in greater compression ratios. In addition, motion compensated interpolative coding scales down the data further through the use of bidirectional prediction. In this case, the encoding takes place based upon the differences between the current frame and the previous and next frames in the video sequence.

MPEG encoding involves three kinds of frames:

1) I-frames (intraframe encoded)
2) P-frames (predictive encoded)
3) B-frames (bidirectional encoded)

As evident from their names, an I-frame is encoded completely as it is without any data loss. An I-frame usually precedes each MPEG data stream. P-frames are constructed using the differences between the current frame and the immediately preceding I or P frame. B-frames are produced relative to the closest two I/P frames on either side of the current frame. The I, P, and B frames are further compressed when subjected to DCT, which helps to eliminate the existing interframe spatial redundancy as much as possible.
A significant portion of the interframe encoding is spent in calculating motion vectors (MVs) from the computed differences. Each non-encoded frame is divided into smaller macro blocks (MBs), typically 16 × 16 pixels. Each MV has an associated MB. The MVs actually contain information regarding the relative displacements of the MBs in the present frame in comparison with the reference. These are calculated by extracting the minimum value of sum of absolute differences (SADs) of an MB with respect to all the MBs of the reference frame. The resultant vectors are also encoded along with the frames. However, this is not sufficient to provide an accurate description of the actual frame. Hence, in addition to the MVs, a residual error is computed, which is then compressed using DCT. It has been proven that the ME and DCT blocks are the most computationally expensive components of an MPEG encoder [10], [15]. The different steps involved in performing MPEG compression are shown in Fig. 1.

There are multiple ways of setting the hard threshold for the output PSNR, which determines whether the quality of a video is acceptable or not. For the sake of simplicity, it is assumed that either the absolute PSNR or the percentage change in PSNR serves as a faithful yardstick for evaluating the quality of videos outputted by the approximated MPEG encoder. In this regard, we define two metrics: 1) absolute error threshold (AET) and 2) relative error margin (REM) to demarcate between the acceptable and unacceptable videos. AET is defined as a fixed absolute PSNR value below which the video is termed to be unacceptable. REM is expressed as a certain percentage of the base PSNR value, which gives the maximum permissible degradation in output PSNR. Either of them can be utilized for judging the merit of a video. In the case AET is fixed at 25 (evaluated by a subjective assessment of the video qualities).

Disadvantages:

- Power consumption is high

Proposed System:

Reconfigurable Adder/Subtractor Blocks:

Dynamic variation of the DA can be done when each of the adder/subtractor blocks is equipped with one or more of its approximate copies and it is able to switch between them as per...
requirement. This reconfigurable architecture can include any approximate version of the adders/subtractors.

Fig. 2. 1-bit DMFA.

The proposed scheme replaces each FA cell of the adders/subtractors with a dual-mode FA (DMFA) cell (Fig. 2) in which each FA cell can operate either in fully accurate or in some approximation mode depending on the state of the control signal APP. A logic high value of the APP signal denotes that the DMFA is operating in the approximate mode. We term these adders/subtractors as RABs. It is important to note that the FA cell is power-gated when operating in the approximate mode. Fig. 2 shows the logic block diagram of the DMFA cell, which replaces the constituent FA cells of an 8-bit RCA, as shown in Fig. 3. In addition, it also
Fig. 5. 8-bit reconfigurable RCA block.

Inclusion of RABs in ME and DCT Blocks

The SAD computational adders along with the internal DCT adders and subtractors are replaced with RABs. Each RAB can be conceptualized as an RCA (or any other adder block) with all the FA cells replaced by DMFAs (or DMCLB and DMPG blocks for CLA).

Advantages:

- Optimize the power consumption

Software implementation:

- Modelsim
- Xilinx ISE