

平成 2 7 年 6 月 2 9 日現在

機関番号：2 5 3 0 1

研究種目：基盤研究(C) (一般)

研究期間：2011 ~ 2014

課題番号：2 3 5 0 0 0 6 4

研究課題名 (和文) 符号の状態に応じて適応的に動的再構成を行うC A B A Cコーデックアーキテクチャ

研究課題名 (英文) Dynamically Reconfiguration CABAC Codec Architecture Adaptive to Code Status

研究代表者

小椋 清孝 (KOMOKU, Kiyotaka)

岡山県立大学・情報工学部・助教

研究者番号：9 0 2 7 5 3 7 1

交付決定額 (研究期間全体) : (直接経費) 4,000,000 円

研究成果の概要 (和文) : H.264における画像のブロック残差信号をターゲットとしたCABAC復号処理について、出現する係数値を予測し、あらかじめ投機的に区間計算を行うことで処理を高速化する方法について検討を行った。

まず、エンコード処理にかかる残差データの符号分布解析を行った。その結果に基づき、"マルチシンボル復号回路"と"1シンボル復号回路"の2種類の回路を並列に配置する構成を提案した。現在、回路の実装を進めている。

研究成果の概要 (英文) : For CABAC decoding process for block residual data in H.264, the high performance decoding architecture is considered, which predicts the residual data value and speculatively calculate interval subdivision of arithmetic decoding.

In first, the code distribution analysis of encoding residual data is investigated. Based on the results, the CABAC architecture which performs a "multi-symbol decoder" and a "1-symbol decoder" in parallel is proposed. The circuit is now implementing.

研究分野：集積回路工学

キーワード：CABAC コーデック H.264 H.264/AVC 復号器

1. 研究開始当初の背景

CABAC(Context-based Adaptive Binary Arithmetic Coding: コンテキストベース適応算術符号化)は、動画符号化標準である H.264/AVC や HEVC において用いられる算術符号化手法である。非常に高圧縮率である反面、逐次性が非常に高い処理アルゴリズムであるためにエンコード・デコードともに計算量が非常に大きくなっている。従来の MPEG などの動画コーデックにおいては、このような計算量の大きな処理をハードウェアで並列化等を行うことにより利用可能としてきたが、現時点において、CABAC ハードウェアの高速化については、多くの試みが行われてきたにもかかわらず、まだ性能面で十分であるとはいえない状況である[1]。

CABAC 処理の性能指標としては、中間表現である 2 値化データの 1 クロックサイクルあたりの処理数 bin/clk が用いられるが、CABAC の基本的な処理アルゴリズムでは $1\text{bin}/\text{clk}$ の処理性能となっている。図 1 に CABAC 符号処理のフローを示す。入力されたシンボル値である多値信号が 2 値化分で 2 値信号に変換され、この 2 値信号 $1\text{bit}(\text{bin})$ ごとに算術符号化およびコンテキスト情報の更新が行われる。この処理の逐次性から処理の並列化が困難とされている。

1 クロックサイクルあたりの処理 bin 数を如何に増やすかという取り組みがこれまでの研究で行われてきた。H.264/AVC では、CABAC 処理はフレームを構成するスライスという単位で行われるため、スライス単位で並列化して高速化するという方法が提案されているが、これ以外では、 1bin を処理するユニットを直列に接続し、1 クロックサイクル中に複数の bin を処理するという方法が主流である。つまり、スライスよりも小さい単位での CABAC 並列処理についてはあまり検討が行われていないという状況である。

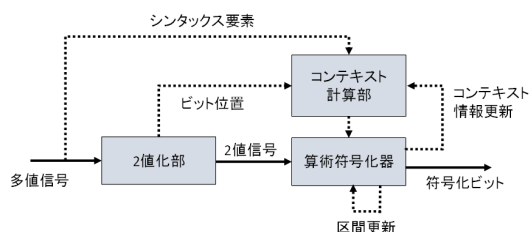


図1 CABAC 符号処理のフロー

2. 研究の目的

本研究では、高速処理可能な CABAC ハードウェアアーキテクチャの開発を行うことである。従来、不可能であるとされてきたス

ライスより小さい単位である処理パラメータ単位での並列処理を可能とするアーキテクチャの検討を行うこととした。

H.264/AVC では、さまざまなパラメータが CABAC 処理の対象となっているが、本研究では、CABAC 処理の大部分を占める量子化後の係数データである残差信号(residual data)を対象とした。動き補償→DCT→量子化といった処理の流れから処理対象のパラメータ値が様々な条件のもとで限定されることを想定し、これを予測して予め CABAC の区間計算等をいくつかの値に対して行っておき、実行時はこれらの候補中から選択することで、高速化が可能となる。当然、予測が外れる場合のペナルティーも存在するが、精度良く出現値を予測して投機実行の成功率を上げることができれば、全体としての性能向上を図ることが可能となるはずである。

さらに、残差データで構成される各マクロブロックの符号化モード等を手掛かりに、出現する値の分布の違い等を判別し、例えば小さい値が頻出するブロック向けの投機実行構成といった実行構成をいくつか用意し、それを CABAC 処理時に適応的に切り替える(動的再構成)ことが可能になればさらに高性能化が期待できると考えた。特に小さい値が頻出する場合、1 クロックサイクルで 1 シンボル(2 値化変換する元の値)実行したとしても、 bin/clk でみればあまり大きな値にならない。このような場合、複数シンボルを一度に処理できる構成をとることで処理性能を上げることができる。一方、大きな値が連続する場合、このような構成は投機予測が外れるリスクが高く、回路規模も肥大化するため好ましくない。

3. 研究の方法

CABAC 処理される残差データのシンボル値の分布解析を行い、それに従って処理回路のアーキテクチャの検討を進めることとした。

残差データのシンボル値の分布解析は、H.264/AVC reference software JM18.5 を用いて、H.264/AVC エンコード時に CABAC 処理関数に渡される残差データを抽出し、そのデータを用いて解析を行った。解析は画像サイズやビットレート等が異なる数 100 フレーム程度の長さの動画データに対して行った。エンコード時に処理されるブロックデータすべてについて解析を行い、ブロック種ごとの「非 0 シンボル数」や「Leading One 長」などブロックを特徴づけられる特徴量について調査した。ブロック種は Luma-Intra 16-DC, Luma Intra16-AC, Luma-4x4, Chroma-DC, Chroma-AC, 8x8 の 6 種類である。

ここで、調査した各特徴量について概要を示す。「非 0 シンボル数」は非 0 の値を持つ各シンボルの個数分布である。「非 0 シンボル総数」は、ブロックに含まれる非 0 シンボルの個数の分布である。「Leading One 長」は各ブロックデータのジグザグスキャン後の先頭に、シンボル '1' が連続して出現する長さである。

符号分布結果を受けて、投機実行により処理を行う CABAC 復号回路のアーキテクチャについて検討を行った。検討したアーキテクチャの実装は、Xilinx 社の FPGA をターゲットにして行った。設計ソフトウェアは Xilinx Foundation ISE、使用した設計言語は VHDL である。

4. 研究成果

解析したデータのうち、suzie_qcif.yuv エンコード時の解析結果を示す。

図 2 に、各ブロック種別における非 0 シンボル数分布の結果を示す。シンボル '1' がほとんどを占め、'2' 以上のシンボルとは出現確率が大きく異なることがわかる。'2' 以上については、'4' までは漸減しているのがかろうじて見て取れるが、それ以上についてはかなり低い出現確率である。ブロック種での違いについては、Chroma-AC でシンボル '1' がほぼ 100% を占めている以外はこれといった特徴はない。

図 3 に、中に含まれる非 0 シンボル個数を軸にしたブロック数の分布を示す。例えば、'1' の部分は処理ブロック総数における非 0 シンボルを 1 個だけ含むブロックの数(割合)を示している。ここでは、非 0 シンボル総数 0 がほぼ支配的であるが、Chroma-DC では個数 1 の割合が個数 0 を上回っているなど、ブロック種別での違いが見て取れる。

図 4 に Leading One 長の分布を示す。Luma-intra 16-DC や Chroma-DC では、他のブロック種と比較すると長さ 0 の占める割合が多い。また、いくつかのブロック種では、長さ 1 以上の出現確率について、長さが変わっても急激に出現確率が減少せず、ほぼ同程度の値を保っているという点が特徴的である。

図 5 に、Leading One 長について、ブロック種 8x8 のときの分布について、異なる qcif サイズの動画データで比較した結果を示す。suzie_qcif.yuv だけ長さ 0 の占める割合が、他と比べて若干大きいものの、特に長さ 1 以上については全体的に動画間での大きな違いは見られない。

以上の解析を、他の動画データについても行ったが、結論として、ブロック種や動画データの違いを明確に反映するブロックデータの分布特徴を得ることができなかった。これにより、対象ブロックの特徴により動的に構成を変化させて処理を行うという当初のアプローチは、ブロックの識別を行う方法がないため、実現が厳しいものとなった。

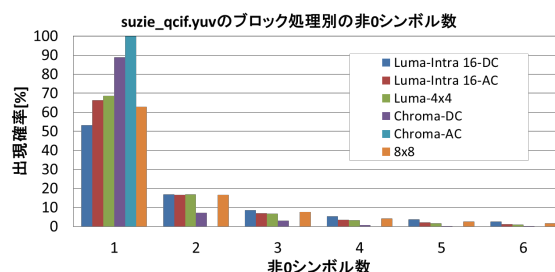


図 2 suzie_qcif.yuv での非 0 シンボル数分布

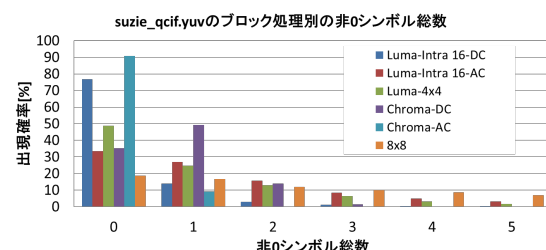


図 3 suzie_qcif.yuv での非 0 シンボル総数分布

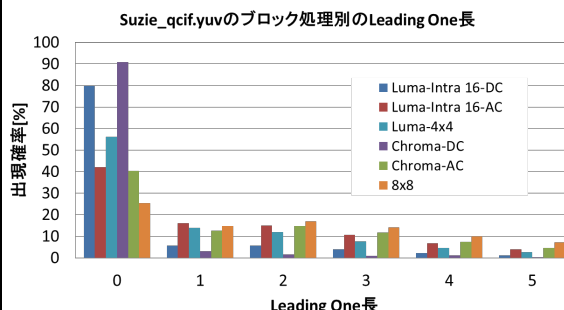


図 4 suzie_qcif.yuv の LeadingOne 長分布

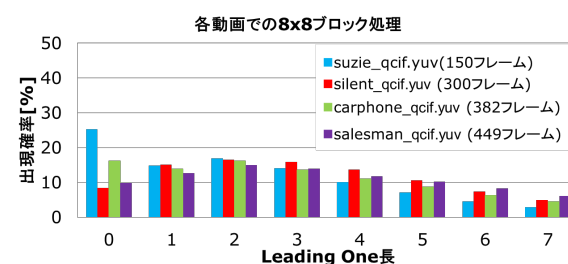


図 5 異なる動画データ間における Leading One 長分布の比較

一方、図 4、図 5 から、Leading One が 0 でない場合は、さまざまな長さがほぼ均等に出現すること、また、この傾向は動画データ間でもあまり差異がなさそうであることから、シンボル '1' が連続して出現することを前提とした Leading One 用の "マルチシンボル復号回路" が有効であると考えられる。シンボル '1' については 2 値化後の bin 長も短く、前もって複数の state 値をたどって区間計算を行っておくことは、より大きなシンボル値と比較すれば難しくないといえる。

これにより、ブロックの最初に連続したシンボル 1 があれば、CABAC 処理量として複数シンボル/clock を実現可能である。

さらにこの回路と並列に、シンボル '1' 以外を処理する通常の処理回路も配置する。なお、残差データの 2 値化後の bin 構成が「連続する 1+0」ということから、この bin 構成を前提とした「1 シンボル復号回路」にすることで、1 シンボル/clock、つまり複数 bin/clock を達成するものとする。

以上の、2 種類の回路を並列に配置する CABAC アーキテクチャを検討し、設計を開始した。1 シンボル復号回路については、正負符号処理および正規化処理を省いたプロトタイプ的设计を行い、基本動作の確認を行った。今後も引き続き実装を進め、提案構成の評価を行う予定である。

<引用文献>

Yahya Jan, Lech Józwiak, "Survey of Advanced CABAC Accelerator Architectures for Future Multimedia," ARC2009, LNCS 5453, pp.342-348, 2009.

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[学会発表](計 3 件)

宮崎雅之, 小椋清孝, 森下賢幸, 伊藤信之, "投機的手法による H.264 CABAC 復号の検討", LSI とシステムのワークショップ 2013 講演資料集およびポスター資料集, pp.160-162, 北九州, May, 13, 2013

宮崎雅之, 小椋清孝, 森下賢幸 z 伊藤信之, "投機的手法を用いた CABAC 復号の検討", 第 15 回 IEEE 広島支部学生シンポジウム, A-73, Nov. 2013

木下巧登, 宮崎雅之, 小椋清孝, 森下賢幸, 伊藤信之, "投機的 CABAC のための H.264 係数データ分布解析", 第 16 回 IEEE 広島支部学生シンポジウム, B-44, Nov. 2014

6. 研究組織

(1)研究代表者

小椋 清孝 (KOMOKU, Kiyotaka)

岡山県立大学・情報工学部・助教

研究者番号: 90275371