

DCT への近似加算適用による JPEG 圧縮の低消費電力化の検討

濱寛之^{1,a)} 請園智玲¹ 佐藤寿倫¹

概要 : JPEG 画像圧縮回路の低省電力化を検討している。JPEG は非可逆圧縮方式を採用しており、圧縮前の画像データと圧縮・展開を経たそれとは完全には一致しない。それら二つの画像間に差があるとしても、人の視覚では認識できないため問題とはならない。この点に着目し、近似計算を利用することで低省電力化を図る。DCT (Discrete Cosine Transform) 内の加算に近似加算器 CMA (Carry-Maskable Adder) を用いることを検討している。近似処理後の画像品質と Nangate 45nm セルライブラリを用いた CMA の評価結果に関して考察する。

1. はじめに

静止画像の圧縮方式として JPEG は依然として利用頻度が高く[1]、消費電力の小さな JPEG 画像圧縮回路は応用される機会が大きいと期待できる。JPEG は非可逆圧縮方式であるため圧縮前と圧縮・展開後で二つの画像データには差が存在することが一般的である。人の視覚では強く認識されない画像の高周波数成分を抑圧している。この点に着目し、本研究では近似計算を利用して JPEG 画像圧縮回路の低消費電力化を検討している。JPEG では DCT で画像信号を空間周波数に変換している。CMA[2,3]を用いて DCT 内の加算を近似的に演算し、回路の低消費電力化を図る。

続く 2 節と 3 節でそれぞれ DCT と CMA を説明する。4 節で評価結果を紹介し、考察を行う。5 節はまとめである。

2. DCT

JPEG では DCT で画像信号を空間周波数に変換している。JPEG の DCT では YCrCb 方式の 8×8 ピクセル四方の 64 個のピクセル値データを 64 段階に周波数分解する。続く量子化で高周波成分が削られる。

画像のピクセル値を S_{xy} 、DCT 変換係数を D_{vu} とすると、DCT は以下の式で表される。

$$D_{vu} = \frac{1}{4} C_u C_v \sum_{x=0}^7 \sum_{y=0}^7 S_{xy} \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16}$$

ここで、 v と u は DCT の 2 次元座標、 x と y は入力画像の 2 次元座標、 C_u と C_v は u または v が 0 のときに $1/\sqrt{2}$ となりそれ以外では 0 である。

本研究では、[4]で提案されている DCT アルゴリズムを採用し C++ で 32 ビット整数を用いて実装された[5]を利用している。

3. CMA

CMA[2,3]は正確に加算するモードと近似的に加算するモードとを備える。近似計算モードでは演算精度を制御す

ることも可能である。

図 1 と 2 はそれぞれ CMHA (Carry-Maskable Half Adder) と CMFA (Carry-Maskable Full Adder) である[2]。桁上げ伝搬加算器と同様に、最下位ビットに CMHA を配置し、残りの上位ビットに CMFA をカスケード接続することで、 N ビット CMA ($N \geq 2$) が構成される。

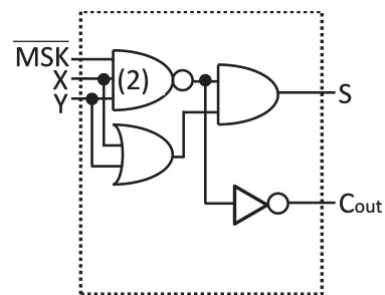


図 1 CMHA[2]

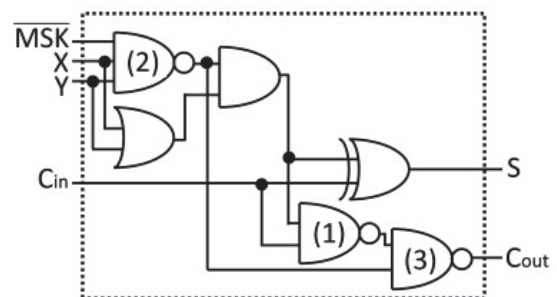


図 2 CMFA[2]

$MSK = 0$ の時、3 入力 NAND ゲート(2)は X と Y を入力とする 2 入力 NAND ゲートとして働く。この時 CMHA では $S = \overline{X} \cdot \overline{Y} \cdot (X + Y) = X \cdot \overline{Y} + \overline{X} \cdot Y = X \oplus Y$ 、 $C_{out} = X \cdot Y$ となるから、CMHA は正確な半加算器として動作する。同様に CMFA では $S = X \oplus Y \oplus C_{in}$ 、 $C_{out} = X \cdot Y + (X \oplus Y) \cdot C_{in}$ となるから、CMFA は正確な全加算器として動作する。

一方 $MSK = 1$ の時、3 入力 NAND ゲート(2)の出力は 1 に固定される。この時 CMHA では $S = A + B$ 、 $C_{out} = 0$ となるから、CMHA は OR ゲートとして機能する。 $C_{in} = 0$ の制約を設けると CMFA も同様に OR ゲートとして機能する。す

1 福岡大学工学部
Fukuoka University, 8-19-1 Nanakuma, Jonan-ku, Fukuoka 814-0180, Japan.
a) tl181280@cis.fukuoka-u.ac.jp

なわち CMHA と CMFA のどちらも近似加算器として動作する。

したがって、最下位ビットから M ビット目 ($N \geq M \geq 1$) までを連続して $MSK = 1$ と設定しなければならないという制約を設けることで、上位 $(N - M)$ ビットが正確な加算で下位 M ビットを OR 演算で近似する N ビット近似加算器として動作する。また M を増減することで演算精度を制御できる。一方 N ビット全てで $MSK = 0$ と設定すれば、正確な N ビット桁上げ伝搬加算器として動作する。

4. 評価

4.1 評価環境

評価には 255×255 ピクセルの lena のカラー画像を用いる。

まず桁上げがマスクされる (近似加算される) ビット数を変化させて、近似的な JPEG 画像圧縮による劣化の度合いを調べる。[5]で公開されている JPEG エンコーダを利用し、DCT 内部の加算を CMA による近似加算に置き換えて画像を圧縮する。圧縮前と圧縮・展開後の二画像を比較して画像品質を評価する。指標には PSNR (Peak Signal to Noise Ratio) と SSIM (Structural SIMilarity) を採用する。評価の目安[6]を表 1 に示す。

表 1 圧縮画像の数値目安[6]

PSNR	SSIM	主観評価
40~∞[dB]	0.98 以上	元の画像と圧縮画像の区別がつかない
30~40[dB]	0.90~0.98	拡大すれば劣化がわかるレベル
30 以下[dB]	0.90 以下	明らかに劣化がわかる

続いて 32 ビット CMA の消費電力を評価する。[5]では 32 ビット整数で DCT が処理されているため、そのように決定した。Verilog-HDL で CMA を実装した。上記の画像品質評価で利用した JPEG エンコーダから DCT 内加算での加数と被加数を抽出し、VCS によるシミュレーションを実施する。lena の 1 画像の処理で、凡そ 65 万回の加算が実行される。その結果をもとに DesignCompiler で消費電力を評価する。合成には 45nm Nangate セルライブラリ[7]を利用した。

4.2 画像品質

図 3 に、圧縮前と圧縮・展開後の二画像を比較した PSNR をまとめる。横軸が桁上げマスクのビット数で、縦軸が PSNR である。劣化の度合いが著しいため、桁上げをマスクするビット数が 7 までの結果を示している。PSNR が 40dB を超えているのは桁上げマスクが 1 ビットの時のみで、2 ビット以上では 40dB を下回る結果となった。桁上げマスクが 4 ビットを超えると PSNR は 30dB を下回り、目視で画像を比較すると明らかに縦方向にノイズが生じていることが確認された。

図 4 に SSIM の結果をまとめる。横軸が桁上げマスクの

ビット数で、縦軸が SSIM である。元画像と区別がつかない目安である 0.98 以上の SSIM となったのは桁上げマスクが 1 ビットの時のみである。4 ビットを超えると SSIM は 0.6 を割っている。PSNR での評価と同様の結果であり、4 ビットを超える桁上げマスクは実用的とは言えない。

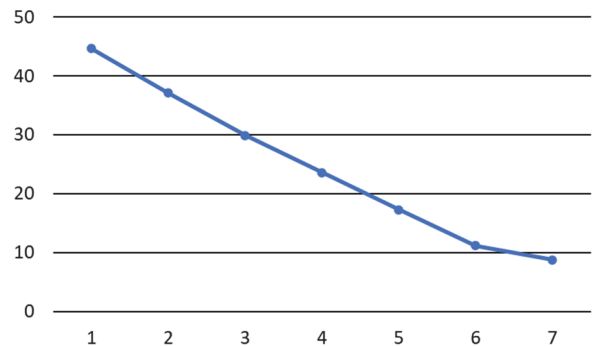


図 3 PSNR (dB)

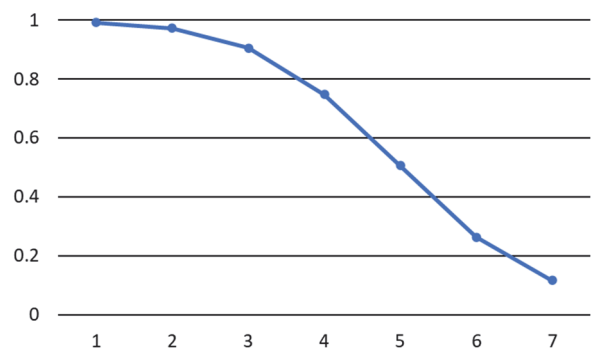


図 4 SSIM

4.3 消費電力

4.2 節の結果に基づき、桁上げマスクビット数を 0 から 4 の範囲で変化させた結果を図 5 に示す。横軸が桁上げマスクのビット数で、縦軸が動的消費電力である。マスクしない場合の消費電力で正規化している。桁上げをマスクすることで消費電力が増大している。[3]での評価結果では桁上げマスクビット数を増やすと線形に消費電力を削減できることが確認されており、期待とは異なる結果となった。

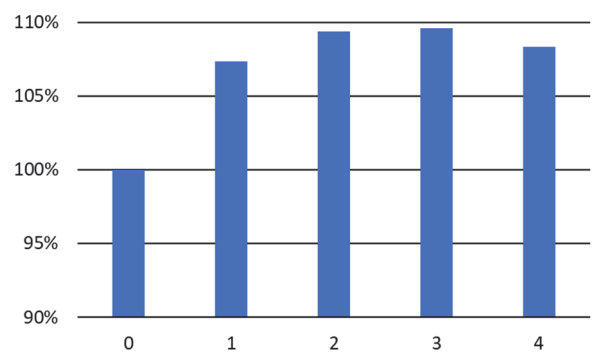


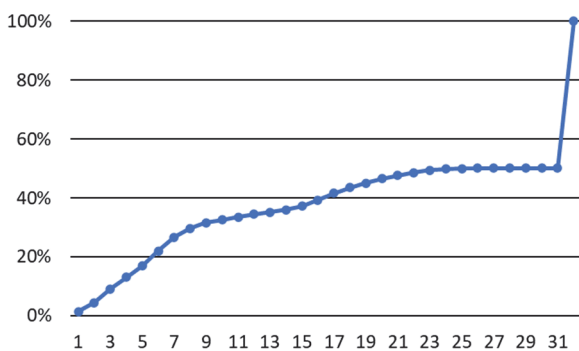
図 5 消費電力

4.4 考察

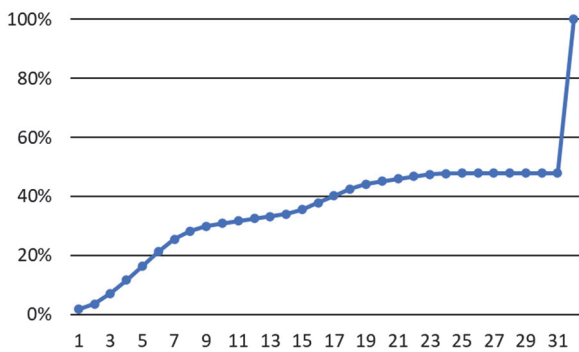
桁上げをマスクすることで、急激に画像品質が劣化する

こと、また消費電力が増大することは、予想外の結果であった。DCT 内加算での加数と被加数が一様ではなく何かしらの偏りを持っていることが原因ではないかと予想し、それらの特徴を調査した。以下では桁上げマスクビット数は 1 である。

図 6 は '1' となる最上位ビットを累積グラフで表現したものである。横軸がビット位置 (1~32) である。(a) が正確な桁上げ伝搬加算器の結果であり、(b) が 32 ビット CMA の結果である。いずれの場合も 32 ビット目がほぼ 50% を占めていることが分かる。すなわち、加算器の入力の半数が負数であり、加えて (b) の方が負数の割合が大きいことが確認された。CMA は負数を苦手としており [8]、このことが画像品質の著しい劣化を引き起こしている原因であろうと思われる。



(a)桁上げ伝搬加算器

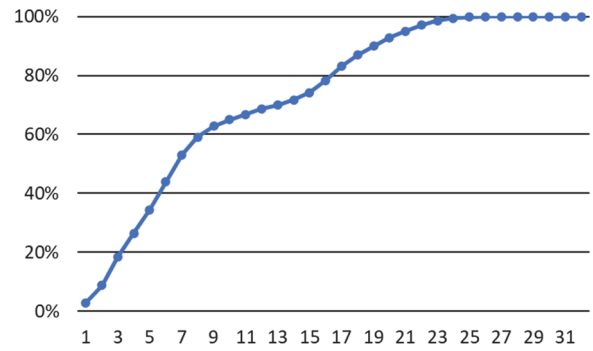


(b)CMA

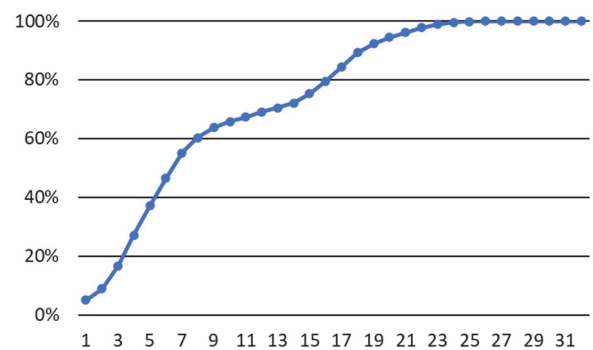
図 6 '1' となる最上位ビット位置

図 7 は図 6 と同じデータを絶対値での有効ビット数 (すなわち、正数では '1' 負数では '0' となる最上位ビット) を累積グラフで表現したものである。横軸がビット数である。いずれの場合でも 8 ビット以下で全体の 60% を超えていることから、絶対値の小さな値が多いことが分かる。図 8 は図 7 のデータを分布図として表現したものである。CMA の結果ではグラフが全体的に左側へ圧縮されており、より小さな値を加算していることが確認される。CMA の方が長い桁上げを生じる可能性が高いと予想され、これが桁上げをマスクした場合に消費電力が大きくなる原因ではないか

と思われる。

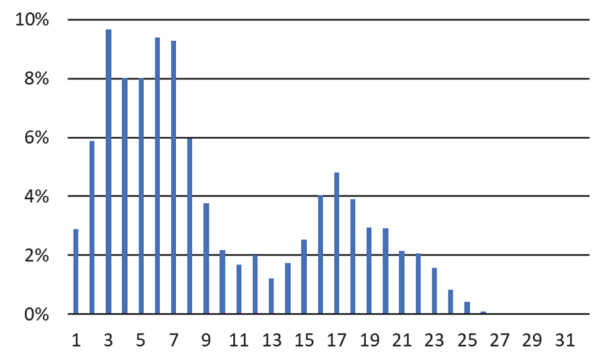


(a) 桁上げ伝搬加算器

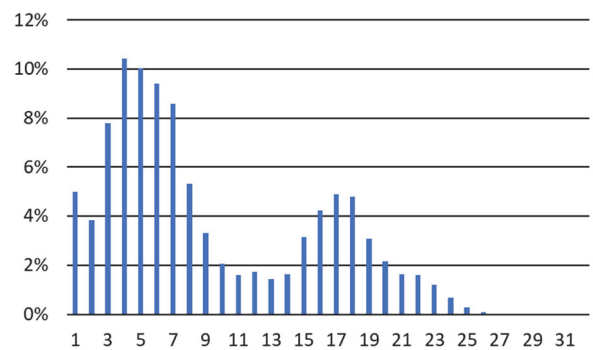


(b)CMA

図 7 有効ビット幅



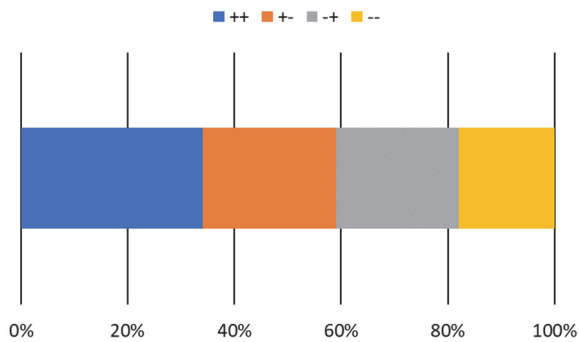
(a)桁上げ伝搬加算器



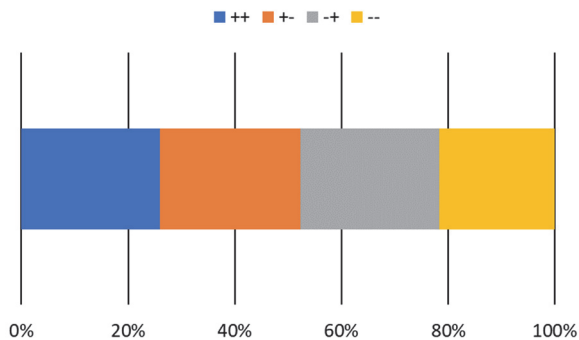
(b)CMA

図 8 有効ビット幅 (分布)

図9には加数と被加数の組み合わせをまとめた。二正数(++), 正数と負数(+), 負数と正数(-), そして二負数(--)の加算の割合を示している。CMAで負数を含む加算が大幅に増えていることが分かる。正数と負数および負数と正数での組み合わせでの加算において、近似加算をすることで桁上げ伝搬が消失し、正確な加算では結果が正数になるはずの場合に誤って負数の結果となっているのではないかと推察される。これにより負数が増大しているのではないかと想像される。



(a)桁上げ伝搬加算器



(b)CMA

図9 加数と被加数の組み合わせ

5. まとめ

低消費電力な JPEG 画像圧縮回路の実現を目指した初期評価結果を報告した。JPEG は非可逆圧縮方式を採用しており圧縮前と圧縮・展開後の二画像は完全には一致しないという点に着目し、DCT 内の加算に近似加算器 CMA を用いて近似計算を利用することで低消費電力化を図っている。評価の結果、僅かなビット数の桁上げマスクですら画像の劣化が甚だしく、また桁上げのマスクにより消費電力が増大することが確認された。DCT 内加算での加数と被加数の特徴を照査したところ、有効ビット数の小さな値が多数を占めており、また近似加算により負数が増大していることが分かった。CMA は符号付き加算を苦手としており、これらのことが期待外れの結果の原因だと推察される。

今後も上記結果の原因調査を継続し、それらを特定することが目標である。また、符号付きの近似加算器[8]を用い

た JPEG 画像圧縮回路の省電力化についても検討したい。

謝辞 議論に参加して下さったシステムアーキテクチャ研究室の皆さんに感謝します。本研究の一部は、福岡大学の研究助成（課題番号：205008）および科学研究費補助金（研究課題/領域番号：20H00590）の助成を受けたものです。

参考文献

- [1] 後藤大地, PNG と SVG, TECH+, <https://news.mynavi.jp/techplus/article/20191203-931434/>, (参照 2022-10-20).
- [2] 田島加織 他, CMA を用いた画像先鋭化処理専用回路の低消費電力化, 情処研報 2018-SLDM-182, 2018.
- [3] T. Yang et al., An Accuracy-Configurable Adder for Low-Power Applications, IEICE Trans. on Electronics, Vol. E103-C, No. 3, doi: 10.1587/transele.2019LHP0002, 2020.
- [4] C. Loeffler et al., A. Practical Fast 1-D DCT Algorithms with 11 Multiplications," International Conference on Acoustics, Speech, and Signal Processing, doi: 10.1109/ICASSP.1989.266596, 1989.
- [5] <https://github.com/richgel999/jpeg-compressor/>, (参照 2022-10-20).
- [6] 小箱雅彦, 電子化文書の画像圧縮ガイドライン, 月間 IM, Vol. 50, No. 5, 2011.
- [7] Silicon Integration Initiative, 15nm Open-cell Library and 45nm FreePDK, <https://si2.org/open-cell-library/>, (参照 2022-10-20).
- [8] T. Sato et al., Correcting Sign Calculation Errors in Configurable Approximations, Asia Pacific Conference on Circuits and Systems, doi: 10.1109/APCCAS47518.2019.8953155, 2019.