

卒業論文

題目

H.265/HEVC符号化単位
高速決定法の研究

指導教員

近藤利夫 教授

2017年

三重大学 工学部 情報工学科
コンピュータアーキテクチャ研究室

飯田拓真 (413802)

内容梗概

近年，4K 映像や 8K 映像にまで対応する技術の開発が進み，動画像の高精細化が進んでいる．この動画像の高精細化に伴い，動画像符号化の圧縮効率の改善が強く求められている．このような状況の中で標準化された H.265/HEVC は圧縮率向上の中核技術として符号化の単位ブロックを多様化している．その符号化単位ブロックである CU，PU は，画像の様が一様な部分で，大きなサイズ， $2N \times 2N$ (正方形) が選択され，動きが多い部分で，小さなサイズ，長方形分割あるいは非対称分割がそれぞれ選択される傾向がある．それにも関わらず，全てのブロックサイズに対してインター予測適用可否判定処理をするため，H.265/HEVC の符号化処理時間増大の主因となっている．これより符号化処理時間の大半を占めているインター予測の適用可否判定処理を低減できればエンコード時間が大幅に短縮される．そこで本研究では，近傍 CU の検出済み動きベクトルを使用し，対象 CU の候補 PU を限定する手法を提案した．また，シミュレーションにより，HM-10.0 のエンコード時間が，限定無しの場合に比べ約 26%低減できることを示した．

Abstract

Recently, as the development of technologies corresponding to high resolution up to 4K video and 8K video progressed, video image resolution has been enhanced. An improvement of compression efficiency is highly demanded due to this enhancement. In this situation, as a core technology of improved coding efficiency, standardized H.265/HEVC diversifies coding block. This structure has a tendency that flat regions are suitable being encoded by large CUs and $2N \times 2N$ (square) PU and complex motion regions are split into small CUs and symmetric or asymmetric portion PU. Despite that, the main reason that encoding time increased is because inter prediction mode decision process is performed for all block size. Encoding time can be remarkably reduced by reducing inter prediction mode decision process which occupy most of encoding time. In this paper, we proposed a fast PU select decision method utilizing detected motion vectors of neighbor CU. The simulation results showed that the proposed method can reduce encoding time by 26% as compared to HM-10.0 without restricting.

目次

1	まえがき	1
1.1	研究背景	1
1.2	研究目的	1
2	H.265 符号化技術と実装上の課題	3
2.1	イントラ予測	3
2.2	インター予測	3
2.3	H.265/HEVC	4
2.4	符号化対象ブロックの領域分割	4
2.5	実装上の課題	6
3	提案手法	6
3.1	方針	6
3.2	近傍 CU の検出済み動きベクトルの分散	7
4	性能評価	10
4.1	実装	10
4.2	評価	10
4.3	考察	12
5	あとがき	14
	謝辞	15
	参考文献	15
A	プログラムリスト	16
B	評価用データ	16

目 次

2.1	PUの種類	5
3.2	対象CUに対する近傍CU	7
3.3	PUの動きベクトル(正方形)	9
3.4	PUの動きベクトル(長方形分割)	9
3.5	PUの動きベクトル(非対称分割)	9

表 目 次

4.1 エンコードする動画像の種類	11
4.2 動画像のエンコード結果	13

1 まえがき

1.1 研究背景

近年，主流のフルハイビジョンの4倍の画素数を持つ4K映像，さらにその4倍の画素数を持つ8K映像にまで対応する技術の開発が進む中，4Kテレビの登場や，スーパーハイビジョンの試験放送開始など，動画像の高精細化が進んでいる．その高精細化に伴う動画像データ量の増大，および符号化に必要な演算量の増加への対応が迫られ，動画像符号化の圧縮効率の改善が強く求められている．

1.2 研究目的

高精細化によるデータ量増大に対応するために，圧縮率が従来 of 2 倍に高められた新しい動画像圧縮規格 H.265/HEVC[1][2] が 2013 年に標準化された．H.265/HEVC では，圧縮率向上の新たな符号化処理単位として，CTU(Coding Tree Unit) と呼ばれる木構造が導入され，最大 64x64 画素サイズから 8x8 画素サイズまで CTU を再帰的に四分木分割するブロック分割構成を採る．分割された木構造の各節を CU(Coding Unit) といい，8x8，16x16，32x32，64x64 サイズを取り得る．さらにその CU を， $2N_x \times 2N_y$ ， $2N_x \times N_y$ ， $N_x \times 2N_y$ ， $N_x \times N_y$ ， $nL_x \times 2N_y$ ， $nR_x \times 2N_y$ ， $2N_x \times nU$ ， $2N_x \times nD$ サイズ

の予測を行う PU(Prediction Unit) と呼ばれるブロックに分割し、インター予測、イントラ予測を行う。また CU は、変換を行うブロックである TU(Transform Unit) にも分割される。領域分割の際に、画像の様相や動きが一様な場合、CTU の四分木構造で階層の高い CU を選択し、画像の様相や動きが複雑な場合、階層の低い CU を選択することで適応的に分割することで圧縮率向上につなげている。この CTU, CU, PU, TU を導入することで、2 倍の圧縮性能を実現し、符号化効率が向上した。しかし、この導入と引き換えに、最適な CU と PU を選択するための処理が膨大になり、符号化に必要な処理量の大幅増加の主因となっている。CU サイズを決定するのに大きな処理量が必要なので、CU サイズの高速決定は、符号化の処理量低減に効果的である。CU 高速決定法としては、輝度値を用いたものや、枝切りなど、多くの提案がなされてきたが、エンコード時間低減率は大きいものの、BD-rate が大きくなってしまいう問題点が生じる。H.265/HEVC では領域分割する場合、画像の様相や動きが一様な場合、CTU の四分木構造で階層の高い CU が選択されるが、階層の低い CU まで全ての候補 CU のコストを算出して、最適な CU を選択する。また候補 CU では、多様な PU 候補に対し、インター予測を試行して、各々の予測モードの差分画像に基づいてコスト計算を行い、最小コストの PU

候補を選択する予測モード決定を行う。この際、画像の様相や動きによって選択される PU のサイズが異なってくるが、全てのサイズを用いて処理を行うため、無駄な試行が発生し演算量が膨大となってしまう。本研究では、このような問題を解決するために、対象 CU 内における評価対象の候補 PU の低減方法のアルゴリズムを考案し、演算量やエンコード時間を削減するプログラムを開発する。具体的には、近傍 CU の検出済み動きベクトルの分散を用いて被写体の動きの複雑性を推測し、対象 CU 内における評価対象の候補 PU の低減手法について追求する。

2 H.265 符号化技術と実装上の課題

2.1 イントラ予測

符号化対象ブロックの近傍の復号済みの画素を用いて空間予測を行い、予測画像を生成する予測手法である。イントラ予測での PU のサイズは、 $2N_x \times 2N_y$ と $N_x \times N_y$ の 2 つのサイズで発生符号量評価を行う。

2.2 インター予測

符号化対象ブロックと類似している箇所を参照画像の探索範囲内から探し(動き探索)、動き補償を行う。インター予測での PU サイズは、 $2N_x \times 2N_y$ の正方形、 $2N_x \times N_y$ と $N_x \times 2N_y$ の長方形、 $nL_x \times 2N_y$ 、 $nR_x \times 2N_y$ 、 $2N_x \times nU$ 、 $2N_x \times nD$

の非対称長方形のサイズで試行する．H.265/HEVC 参照ソフトウェア HM では，符号化処理時間の内インター予測の適用可否判定処理が約 60% から 70%を占めている [3] と報告されており，高速化の障害となっている．

2.3 H.265/HEVC

H.265/HEVC は動画像符号化の国際標準規格であり，個々の要素の改良の積み重ねによって，H.264/AVC の 2 倍の圧縮性能を実現した．H.264/AVC では符号化ブロックは 16x16 画素の固定サイズであったが，H.265/HEVC では最大 64x64 画素に拡張されている．H.264/AVC の PU では，正方ブロックを同じ面積の 2 つの長方形に分割していたが，H.265/HEVC では同サイズの長方形以外に 1 対 3 の面積比で非対称な 2 つのブロックに分割する方法も採用されている．この多様化により，以前より効率的に領域分割し，最適な CU サイズを選択することができる．しかし，符号化ブロックの多様化を採用したことで，圧縮率の向上に繋がったものの，符号化処理量が増加してしまうという問題が生じている．

2.4 符号化対象ブロックの領域分割

動画像符号化では，各画像がブロック単位で分割され，ブロック毎に処理される．H.265/HEVC では，H.264/AVC でのマクロブロックに代わり，

CTU と呼ばれる最大 4 階層の符号化ブロックを四分木分割により階層的に分割することができる。その CTU を再帰的に四分木分割した節を CU と呼び、画像の特性を利用して多様な大きさの CU に分割して符号化処理することができる。また、CU は PU に分割することができ、図 2.1 のように $2N \times 2N$ 、 $2N \times N$ 、 $N \times 2N$ 、 $N \times N$ 、 $nL \times 2N$ 、 $nR \times 2N$ 、 $2N \times nU$ 、 $2N \times nD$ サイズといった多様な PU 候補に対し、インター予測やイントラ予測の符号発生量を評価している。CU を様々な PU で動き補償を試行することで、動き部分が微少な場合に有効である。画像の様が一様な場合、CU は大きなサイズ、PU は $2N \times 2N$ (正方形) が選択され、動きが多い部分には CU は小さなサイズ、PU は長方形分割や非対称分割のものを最適なサイズとすることができ、符号化効率が向上した。

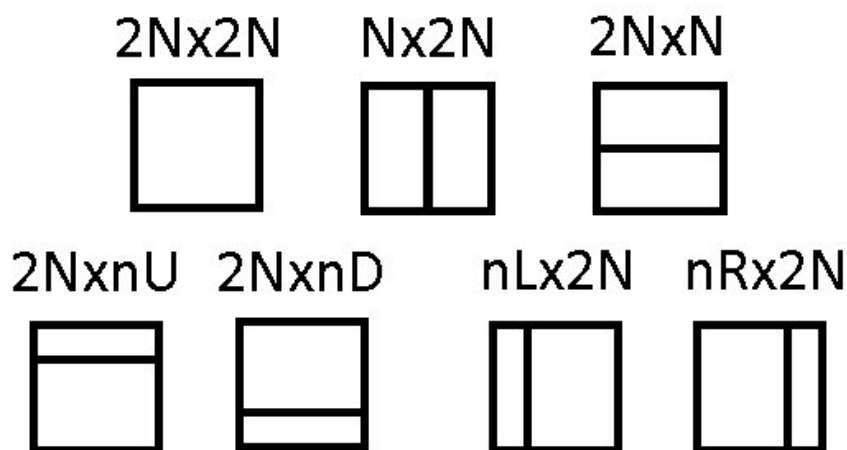


図 2.1: PU の種類

2.5 実装上の課題

圧縮率が向上した反面，最適な CU と PU を選択するための処理が膨大になり，無駄な試行が発生し，符号化に必要な処理が増加している。最適な CU のサイズが大きいサイズであっても，階層の低い CU を試行して最適サイズを決定するので無駄な試行が多い。採用される CU，採用されない CU でも多様な PU で動き補償を試行するので，さらに無駄な試行が多くなる。画像の動きや模様などの特性を活かして，適応的に CTU の枝切りによる高速化や評価対象となる候補 PU を低減できれば演算量を大幅に減らせる。

3 提案手法

3.1 方針

CU サイズを早期に決定し，符号化処理を高速化する手法が提案されているが，符号化時の QP によって高速化性能が大きく変わる。この手法では QP が小さい場合でのエンコード時間削減率は 10%程であり，全体でのエンコード時間削減率が増えない。そこで，画像の動きベクトルの大きさを用いた手法に着目した。近傍 CU の検出済み動きベクトルの分散を用いて被写体の動きの複雑性を推測し，対象 CU 内における評価対

象の候補 PU の低減手法を提案する。

3.2 近傍 CU の検出済み動きベクトルの分散

対象 CU 内における PU 分割は，対象 CU 内の被写体の動きの複雑性によって推測できると考えられる．そこで，図 3.2 のように対象 CU の CULeft(左)，CUAboveLeft(左上)，CUAbove(上)，CUAboveRight(右上)の符号化済み 4 近傍 CU の検出済み動きベクトルから，対象 CU の被写体の動きの複雑性を推測する．対象 CU を処理する際，近傍 CU では既に

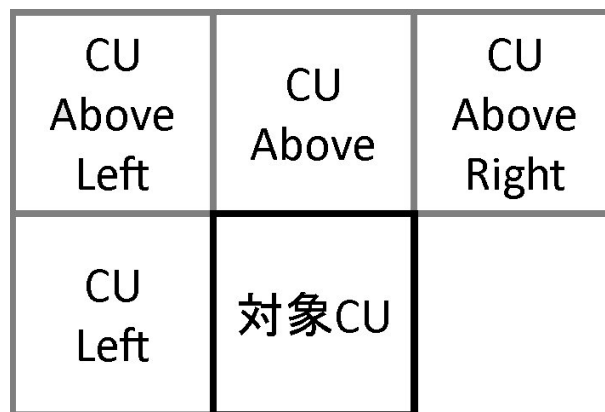


図 3.2: 対象 CU に対する近傍 CU

最適な CU サイズや PU サイズが決定され，被写体の動きを表す動きベクトルも検出済みである．そのためこれらの動きベクトルから被写体の動きの複雑性を表す指標として分散を計算する．しかし，動きベクトルは $2N \times 2N$ (正方形) の場合は図 3.3 のように 1 本，長方形または非対称分

割の場合は図 3.4, 図 3.5 のように 2 本あり, それぞれの CU が選択した PU により数が異なる上に分割面積も異なってくる。そこで, 分散を計算する際に面積比において重み付けをする。具体的には, $nLx2N$, $nRx2N$, $2NxnU$, $2NxnD$ に分類される非対称 PU では, CU を 1:3 に分割するので, この面積比 1 の部分を動きベクトル 1 本分とする。非対称 PU の面積比 3 の部分は 3 本分, $2NxN$, $Nx2N$ の長方形分割では 2 本分, $2Nx2N$ の正方形分割では 4 本分として分散を計算する。例えば, 図 3.5 のように $nLx2N$ で分割された場合は 1:3 の面積比になるので, 左側の動きベクトルは等倍, 右側の動きベクトルは 3 倍と計算する。動きベクトルの水平, 垂直成分それぞれで分散を計算し, これらの値がいずれも閾値以下の場合, 近傍のブロックは同一被写体であるか, カメラ固定あるいはパンであり動きが類似していると推測されるので, 対象 CU も同じ動きであると予想される。その場合, 対象 CU 内を被写体の動きに合わせて更に分割する必要が無いと推測できるため, $Nx2N$, $2NxN$, $nLx2N$, $nRx2N$, $2NxnU$, $2NxnD$ などの長方形分割を候補 PU から除外し, $2Nx2N$ に限定することができる。これにより, 対象 CU でのインター予測の試行を大幅に低減でき, エンコード時間低減に繋がる。

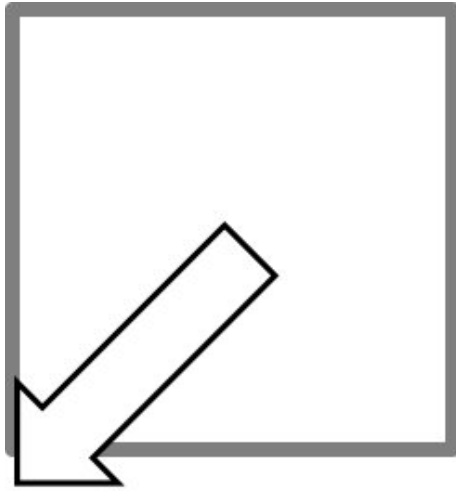


図 3.3: PU の動きベクトル (正方形)

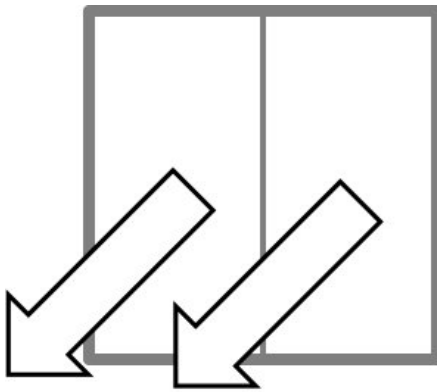


図 3.4: PU の動きベクトル (長方形分割)

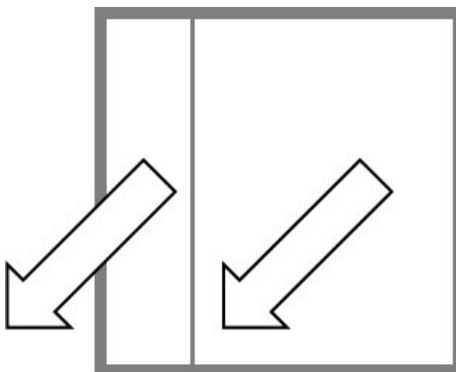


図 3.5: PU の動きベクトル (非対称分割)

4 性能評価

4.1 実装

HEVCの参照ソフトウェアであるHM-10.0に提案手法を実装する。符号化の方式としては、Random Access 符号化を用いている。Random Access 符号化はLow Delay 符号化と異なり、過去のフレームからだけでなく未来のフレームも参照する。そのため、高い符号化効率を得ることのできる符号化方式である。HMのCUサイズを決定するソースであるTEncCU.cppのxCompressCU関数にプログラムを加えていく。符号化対象CUの動きベクトルを用いる場合は、インター予測の各サイズ毎に符号化対象CUの先頭アドレスを取得するgetPartIndexAndSize関数と、そのアドレスを用いてgetMvField関数を用い、動きベクトルを取得する。取得した動きベクトルを垂直成分と水平成分それぞれで平均と分散を求める。近傍CUの場合は、最適PUサイズが確定しているため、xCompressCU関数の先頭付近で符号化対象CUと同じ方法で動きベクトルを取得する。

4.2 評価

HEVCの参照ソフトウェアであるHM-10.0に提案手法を実装し評価を行う。評価は、BD-rate(同一画質を実現した場合の符号量増加率)とエンコー

ド時間低減率を表す ETRR(Encoding Time Reduction Rate) と動き補償
 試行回数低減率を表す MCTRR(Motion Compensation Trials Reduction
 Rate) を標準の HM と比較した . なお ETRR と MCTRR は式 (1) と式 (2)
 により算出する .

$$ETRR = \left(1 - \frac{ET_{Proposed}}{ET_{HM-10.0}}\right) \times 100 \quad (1)$$

$$MCTRR = \left(1 - \frac{MCT_{Proposed}}{MCT_{HM-10.0}}\right) \times 100 \quad (2)$$

評価は , HEVC 共通実験に用いられるシーケンスの中から BasketballDrill(Basket) ,
 BQMall , Kimono , ParkScene(Park) を選択し , 連続する 25 フレームを
 エンコードすることで行った . 使用する画像の解像度 , フレームレート ,
 符号化するフレーム数は表 4.1 に示す .

表 4.1: エンコードする動画の種類

シーケンス	解像度	フレームレート	フレーム数
Basket	832x480	50	25
BQMall	832x480	60	25
Kimono	1920x1080	24	25
Park	1920x1080	24	25

4.3 考察

評価結果を表 4.2 に示す．提案手法は BD-rate の悪化を最大でも 0.8% に抑えつつ，動き補償試行回数は 29.0% 低減できた．更に，エンコード時間は 26.2% 低減することができた．特に Basket と Kimono と Park においては標準の HM に対する BD-rate の悪化率が 1% 未満と良好である．しかし BQMall では 1% 以上の悪化が見られる．これは，BQMall のシーケンスでは被写体が複数動いており，またカメラの動きもあるために，複雑な動きとなっているからだと予想される．そのため，近傍 CU のみの動き情報を用いる提案手法では，対象 CU 内に近傍とは異なる被写体がある場合に局所的な動きの特徴に対応できず BD-rate が悪化したと考えられる．

表 4.2: 動画像のエンコード結果

		提案手法		
シーケンス	QP	BD-rate(%)	MCTRR(%)	ETRR(%)
Basket	22	0.8	29.5	26.3
	27		29.3	26.7
	32		27.6	26.9
	37		26.3	25.3
BQMall	22	1.7	33.1	26.1
	27		33.2	27.7
	32		33.7	29.9
	37		33.3	29.5
Kimono	22	0.3	16.9	18.6
	27		17.8	19.2
	32		18.8	20.3
	37		19.3	20.5
Park	22	0.4	34.3	26.3
	27		35.9	29.3
	32		37.0	32.2
	37		38.4	33.6
Ave		0.8	29.0	26.2

5 あとがき

本研究では、近傍の符号化済み CU の動きベクトルの分散を用いて対象 CU 内の動きを推定し、適応的に候補 PU を限定することで BD-rate の悪化を最小限に抑えつつ、動き補償試行回数を 29.0%、エンコード時間を 26.2%低減できることを示した。このように低減できたのは、近傍の符号化済み CU の動きベクトルを用いた動き予測が効き、対象 CU 内の PU でのインター予測適用可否判定処理の回数が低減されたことによる。BD-rate の悪化も 1%未満に抑えることができたが、動きが複雑なシーケンスでは BD-rate の 1%以上の悪化が見られた。これは、被写体が複数動いており、またカメラの動きもあるために、複雑な動きとなっているからだと予想される。今後は、近傍 CU の動きベクトルのみではなく、対象 CU の動き情報や画像の特徴を用いるなどの様々な手法を組み合わせることにより一層の高速化を試みる。また、全てのフレームでエンコード時間が削減できるように改善し、リアルタイム符号化などの低遅延用途に対する有効性も検証する。

謝辞

本研究を行うにあたり，日頃から様々なご指導をいただきました近藤利夫教授，佐々木敬泰助教，深澤研究員に感謝いたします．また様々な面で世話になったコンピュータアーキテクチャ研究室の皆様に感謝いたします．

参考文献

- [1] 大久保榮，鈴木輝彦，高村誠之，中條健，H.265/HEVC教科書，ISBN-978-4-8443-3468-2，2013年10月．
- [2] G.J.Sullivan，J.R.Ohm，W.J.Han and “T. Wiegand，Overview of the High Efficiency Video Coding (HEVC) Standard”，IEEE Trans. on Circuits And Systems For Video Technology，Vol.22，No.12，Dec 2012．
- [3] J.Vanne，M.Viitanen，T.D.Hmlinen，and A.Hallapuro，“Comparative Rate-Distortion-Complexity Analysis of HEVC and AVC Video Codecs”，IEEE Trans. on Circuits And Systems For Video Technology，Vol.22，No.12，Dec 2012．

A プログラムリスト

H.265/HEVC のテストモデルである HM ソフトウェアエンコーダの HM-10.0 に提案手法を実装した .

B 評価用データ

BasketballDrill , BQMall , Kimono , ParkScene の 4 種のシーケンスを 評価用データとして用いた .