

卒業論文

題目

H.265対応ブロックサイズ適応型
動き検出アルゴリズムの研究

指導教員

近藤利夫

2015年

三重大学 工学部 情報工学科
計算機アーキテクチャ研究室

西村隆太郎 (411840)

内容梗概

2016年のスーパーハイビジョンの試験放送開始が予定されるなど、動画像の高精細化が大きく進んでいる。この高精細の動画像の普及には、PC上でのスムーズな符号化処理が必要不可欠であり、その符号化処理の大半を占める動き検出の高速化が強く求められている。その中で提案された動き検出法のTZSearchは、高検出精度と演算量低減を両立する点で定評があるものの、局所最適解への捕捉を回避するために、まばらな遠隔点の割合が多い拡大型ダイヤモンド探索を繰り返すことから、演算量低減の要求を十分には満たせていない。そこで本研究では拡大型ダイヤモンド探索の遠隔探索を適応的に省く切り替え型ダイヤモンド探索と、H.265/HEVCで採られたブロックサイズの多様化により生じた、異なるサイズのブロック間の探索特性の違いを活かしたブロックサイズ適応型探索を組み合わせた高効率の動き検出法を提案する。また、この提案法を用いることで従来のTZSearchと比べAD演算量を約80%低減できることを示す。

Abstract

Such as test broadcasting of Ultra High Definition is scheduled to start in 2016, high definition of the moving image is enhancing greatly. Smooth encoding process on PC is the essential to spread this high definition image, and speeding up of the motion estimation holding the majority of the coding process is required. Although TZSearch that is proposed under the circumstance has established reputation for compatible with high detection accuracy and low complexity, it can not meet the requirement of reduction complexity because it repeat an expanding diamond search that has many remote search points and these are sparse to avoid capture of local optimal solution. Therefore, this studies present a high efficiency motion estimation method that is combined two search methods. One is a switching-type diamond search that adaptively reduces remote search of expanding diamond search. The other is a block-size-adaptive search that make use of search characteristics with each unit size of various prediction unit size taken in H.265. And show this proposed method can reduce about 80% AD calculation compared to TZSearch.

目次

1	はじめに	1
1.1	背景	1
1.2	研究目的	1
2	動き検出とその高速処理法	3
2.1	動き検出の概要	3
2.2	可変ブロックサイズとH.265における拡張	4
2.3	高速処理法	6
2.3.1	代表的な手法	6
2.3.2	TZSearch	7
3	TZSearchの問題点	11
3.1	探索特性	11
3.2	問題点	12
4	提案手法	14
4.1	方針	14
4.2	切り替え型ダイヤモンド探索	14
4.3	ブロックサイズ適応型探索	16
4.3.1	ブロックサイズ 64x64, 32x32	16
4.3.2	ブロックサイズ 16x16, 8x8	19
4.4	全体のフローチャート	20
5	性能評価	21
5.1	評価	21
5.2	考察	24
6	おわりに	27
	謝辞	28
	参考文献	28
A	プログラムリスト	29
B	評価用データ	29

目 次

2.1	SAD 演算の例	3
2.2	多様なブロックの割り当て	5
2.3	拡大型ダイヤモンド探索の探索点	7
2.4	2点探索の探索点	8
2.5	ラスト探索の探索点	9
3.6	ブロックサイズ別の予測ベクトル誤差	11
4.7	スモールダイヤモンド探索の探索点	15
4.8	切り替え型ダイヤモンド探索の例	15
4.9	切り替え型ダイヤモンド探索のフローチャート	16
4.10	2画素精度サブサンプリング	18
4.11	4画素精度サブサンプリング	18
4.12	16x16 と 8x8 の近傍探索点	19
4.13	全体のフローチャート	20

表 目 次

2.1	H.265 でのブロックサイズ	4
5.2	テストシーケンス	21
5.3	エンコード条件	21
5.4	TZSearch の結果	22
5.5	提案手法の結果	23
5.6	評価結果	23
5.7	テストシーケンスの予測誤差	24
5.8	テストシーケンスの動きベクトルの割合	25

1 はじめに

1.1 背景

普及価格帯の4Kテレビの登場，2016年のスーパーハイビジョンの試験放送開始予定など，動画像の高精細化が大きく進んでいる．高精細動画像の普及には，PC上でのスムーズな符号化処理が必要不可欠である．このため符号化処理の大半を占め高速化の要である動き検出処理の高効率化の研究が継続的に行われてきた．その中で数多くの動き検出アルゴリズムが開発されてきたものの，演算量の低減に加え，転送量の低減の要求までも満たすものはほとんど実現されていない．高検出精度と演算量低減を両立する点が定評され，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で使用されているTZSearchでも，遠隔点の探索が多く探索点がまばらになっているため画像データの転送量低減の要求を十分に満たせてはいない．

1.2 研究目的

高検出精度を維持しつつ，更なる演算量，データ転送量低減の要求に応えるためには，従来のアルゴリズムに加え更なる改良が必要である．そこで当研究室ではH.265/HEVCで採られたブロックサイズの多様化によ

り生じている，異なるサイズのブロック間の探索特性の違いを活かすことで，高検出精度と演算量低減との両立を図る．具体的には，それぞれのブロックサイズ毎に動き探索での特性を調査し，サイズ別に適切な探索法を適用する新たな H.265 対応高効率動き検出アルゴリズムを提案する．

2 動き検出とその高速処理法

2.1 動き検出の概要

動き検出とは一般的に対象画像と参照画像間でのブロックマッチングにより求める，画素の差分絶対値和である SAD(Sum of Absolute Differences)，あるいは SATD(Sum of Absolute Transformed Differences) 値が最小になる点に向かうベクトルを，動きベクトルとして検出する処理のことである．

図 2.1 に SAD 演算の例を示す．この値が小さいほど画像の類似度が高いと判定することができる．なおこの SAD 演算が動き検出での大半の処理となる．

15	14	16	17
20	10	11	15
13	18	12	11
12	9	12	16

12	17	11	19
19	12	15	14
13	9	8	13
21	14	13	16

$$|15-12|+|14-17| \dots +|12-13|+|16-16| = 51$$

図 2.1: SAD 演算の例

検出した位置のブロックを予測ブロックとし，その予測ブロックと符

号化対象ブロックとの差分を符号化対象とする動き補償により，情報量が低減され，圧縮効率が高められている．動き検出は高い精度を得ようとする，従来，符号化処理の大半を占めるほどの膨大な処理量になるため，高精細な動画の普及に向けて，演算量低減に向けた改良が必要不可欠となっている．

2.2 可変ブロックサイズと H.265 における拡張

H.265[1][2] とは 2013 年に標準化された動画符号化規格であり，H.264 の約 2 倍の圧縮率を持つ 4K やスーパーハイビジョン等の超高精細画像に対応する．H.265 と H.264 での動き検出における仕様の大きな違いとして，ブロックサイズの多様化が挙げられる．H.264 では 4×4 - 16×16 であったのに対し，H.265 では $8 \times 4(4 \times 8)$ - 64×64 へと拡張された．またブロックの形状についても，H.264 では正方形を半分に分割した長方形のみ選択可能であったが，H.265 では正方形を 1:3 に分割したような長方形も選択可能である．

表 2.1: H.265 でのブロックサイズ

	H.264	H.265
ブロックの大きさ	4×4 - 16×16	$8 \times 4(4 \times 8)$ - 64×64
ブロックの形状	正方形と長方形	正方形と長方形 (非対称な分割可能)

このブロックサイズの多様化により，動画像の背景など模様が一樣な部分には大きなブロック，細かな模様などがある部分には小さなブロックを割り当てて符号化する．これにより情報量が少ない部分は大きなブロックで一度に処理したりすることで，圧縮率を高めることを可能としている．しかし H.265 のこのブロックサイズの多様化により，圧縮率の向上と引き替えに，従来の H.264 に比べ処理の複雑性や演算量が増している．
図 2.2 では実際に動画像に割り当てられるブロックの例を示す．

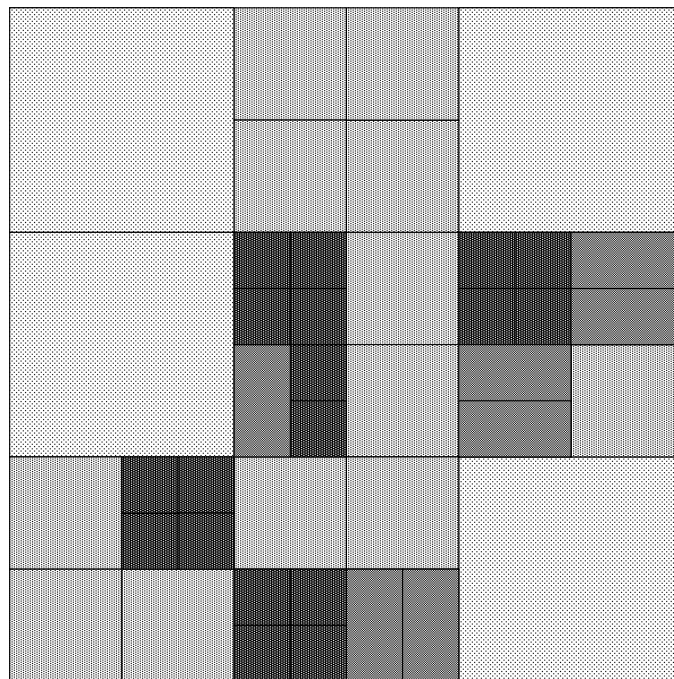


図 2.2: 多様なブロックの割り当て

2.3 高速処理法

2.3.1 代表的な手法

探索範囲の全箇所を探索するような全探索では正確に動きを検出できるかわりに、演算量が膨大となってしまう問題点がある。この問題を解決するために開発された代表的な高効率動き検出アルゴリズムの例として、H.264/AVCの参照ソフトウェアであるJMで使われているUMHexiagonSやEPZSなどがある。これらのアルゴリズムの共通点としては、予測ベクトルを用いて探索開始点を決定し、そこから追跡型の探索方法で動きを検出するという方法である。この追跡型の探索というのは探索パターンの中の最良点に探索中心を移動し再び探索を行う方法で、現在の主流となっている探索方法である。しかし予測ベクトルの近傍の探索には適しているものの、動きが速い場合などに遠隔点の探索がおろそかになり局所最適解に陥ってしまう可能性がある。この場合誤った動きベクトルを検出し続けてしまい、符号化効率および画質に影響が出てしまう。このように様々な動き検出アルゴリズムが開発されているものの、演算量低減と画質や符号化効率の向上を図るのは非常に難しく、それらを両立しているアルゴリズムは数少ない。

2.3.2 TZSearch

TZSearch[3] とは 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で使用されている高検出精度と高効率を両立している数少ない動き検出アルゴリズムの一つである。

主な手法としては予測ベクトルなどを用いて探索開始点を決定し、そこから 2 の累乗の距離の点を探索点とする拡大型ダイヤモンド探索を行っている。図 2.3 に、この拡大型ダイヤモンド探索において、距離 8 まで拡大後の探索点の例を示す。

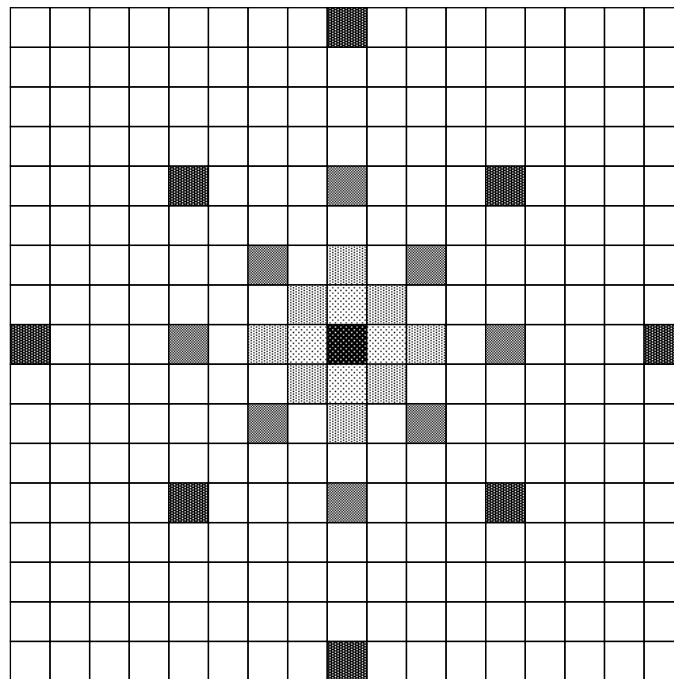


図 2.3: 拡大型ダイヤモンド探索の探索点

なおこの探索の拡大は探索範囲の境界まで行っていく。

TZSearch では拡大後の探索点の中の最良点に探索中心を移動し，拡大型ダイヤモンド探索を繰り返し行い，動きが存在する箇所を追跡することにより，局所最適解に陥ることを避けて高検出精度を確保している．近傍のみの探索の場合，動きが遠隔にあったとしても追跡がそこまでたどりつく前に，局所最適解に陥って探索を終了してしまうからである．

更に探索中心から距離 1 の点が最良点であれば，探索を行っていない距離 2 の点を 2 箇所探索する 2 点探索を行っている．

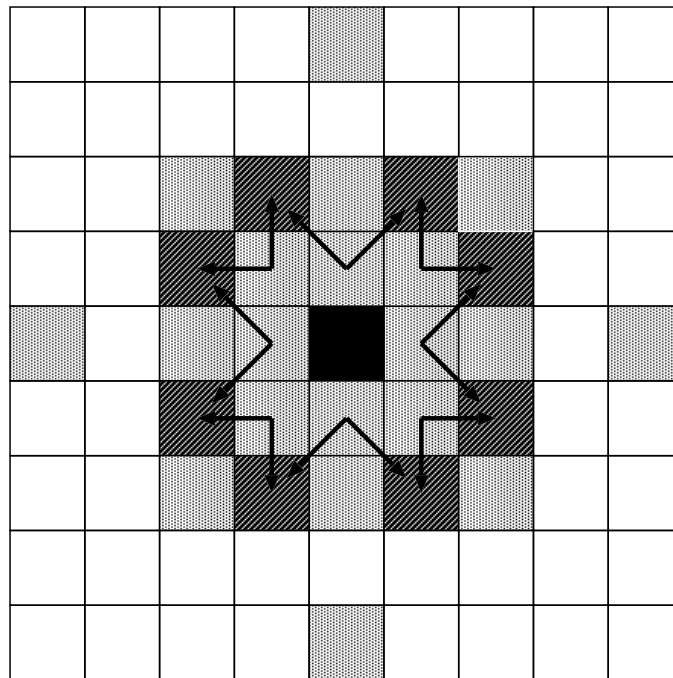


図 2.4: 2 点探索の探索点

この2点探索により探索の早期打ち切りを行う。2点探索では拡大型ダイヤモンド探索の探索中心から距離2の点を探索するので、検出がある程度終了に近づいていることから、打ち切りを行っても精度に影響は少ない上に演算量を低減できる。

また初回の拡大型ダイヤモンド探索での最良点が探索中心から遠い場所にある場合、その最良点が適解ではないと判断して、再び拡大型ダイヤモンド探索を行う前にラスタ探索を行っている。

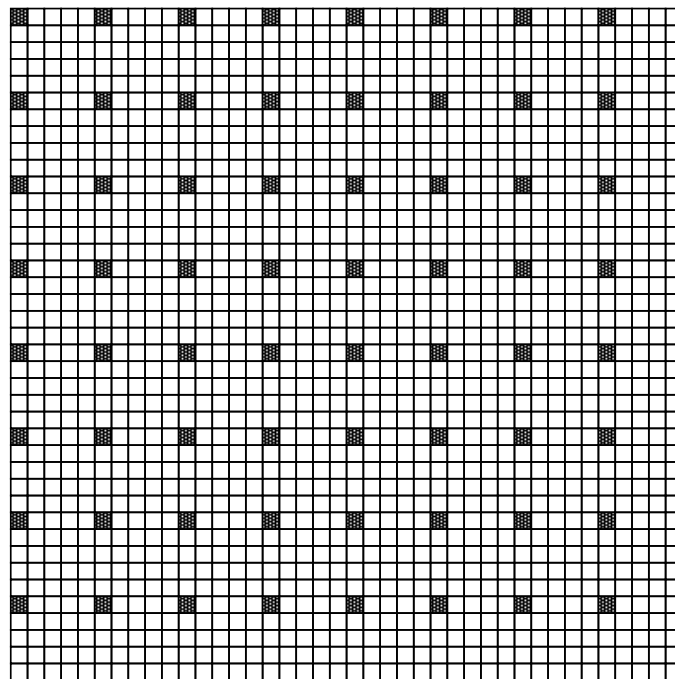


図 2.5: ラスタ探索の探索点

このラスタ探索では探索範囲内を5画素間隔など間隔を開けてブロック

マッチングを行っていく．これにより探索範囲全体を粗く探索していく．

このラスタ探索により動きが速い場合などの精度の維持を図っている．

このように TZSearch では，拡大型ダイヤモンド探索，2点探索，ラスタ探索を組み合わせることで高効率と高検出精度を実現している．

3 TZSearchの問題点

3.1 探索特性

TZSearchにおいて予測ベクトルと検出されるベクトルとの距離である予測誤差について、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でシミュレーションを行った。この予測誤差が小さければ小さいほど、最初の探索開始点の近傍で動きが検出されていることになる。図3.6にtouchdown_pass[4]から抜粋した33フレームをテストシーケンスとして調査を行ったものを示す。

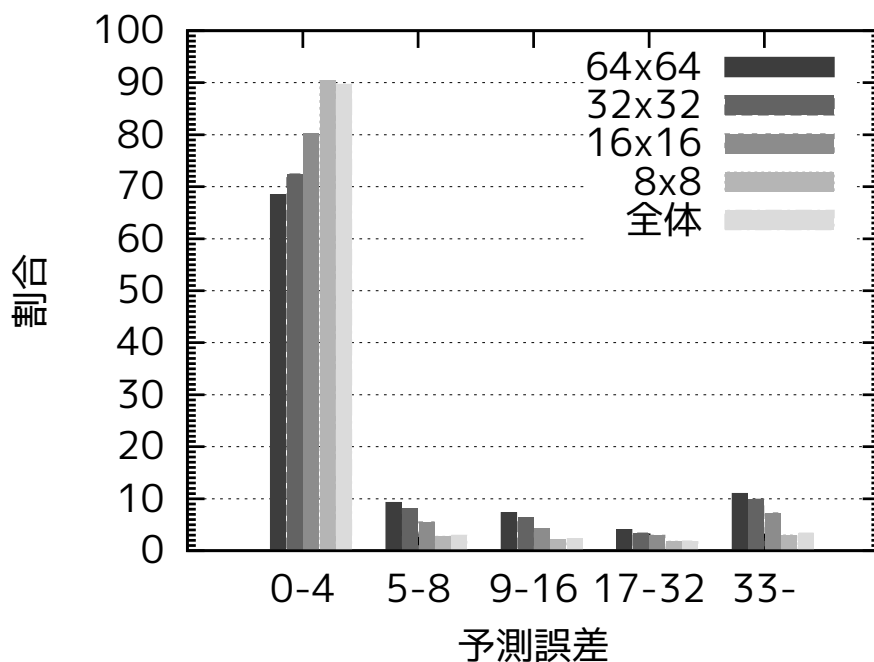


図 3.6: ブロックサイズ別の予測ベクトル誤差

この結果，予測ベクトルで指定される点から遠く離れた点が発見されることはわずかであり，誤差が 4 以内に収まる割合が全体の約 90%を占め，予測ベクトルの精度が非常に高いことが分かる．更にブロックサイズが小さいほどより近傍で発見され，8x8 などのブロックでは 90%以上距離 4 以内で発見される．更に複数のシーケンスでもシミュレーションした結果，これらの特性は動きが速い部分が多々あるシーケンスでも同様であることが明らかになった．

3.2 問題点

予測ベクトルの近傍で動きが発見されることが約 90%を占めるにも関わらず，TZSearch は拡大型ダイヤモンド探索を用いるために，遠隔点を探索する割合が比較的高い．また TZSearch では H.265 でのブロックサイズの多様化を十分に活かさず，全てのブロックサイズで同様の探索法を適用しているので，小さなブロックがより予測ベクトルの近傍で発見されるという特性を活かせていない．これらから TZSearch での遠隔点の探索する割合を小さなブロックほど低減することができ，それにより演算量を大幅に低減できると考えられる．

また TZSearch の遠隔点まで拡大を続ける拡大型ダイヤモンド探索では，

遠隔の探索点がまばらになるため、参照画像データの再利用性が悪化してしまいう問題点もある。この問題点についても小さなブロックほど遠隔点の探索を低減することで、大幅に改善されると考えられる。

4 提案手法

4.1 方針

三章で示した TZSearch の問題点をふまえて，遠隔点の探索を行いながらもより近傍を効率的に探索できるようにする．更に従来の TZSearch では全てのブロックについて同様のアルゴリズムを適用していたが，ブロックサイズが小さいほど遠隔点の探索を省く割合が高くなるようにする．また，ブロックの構成画素数の多い，大きなブロックについてはサブサンプリングによる演算量低減も図る．

4.2 切り替え型ダイヤモンド探索

切り替え型ダイヤモンド探索では，拡大型ダイヤモンド探索での最良点が探索中心から距離 4 以内の場合，続いてスモールダイヤモンド探索を 2 回まで行う．

各探索で，探索中心が探索パタン内の SAD 最小点に一致すればその探索で終了する．2 回目のスモールダイヤモンド探索でも終了しない場合は再度拡大型ダイヤモンド探索からやり直す．ここで，スモールダイヤモンド探索とは探索中心の上下左右 4 点のみからなるスモールダイヤモンドパタンを探索する手法で，この 4 点に SAD 最小箇所があれば探索中心

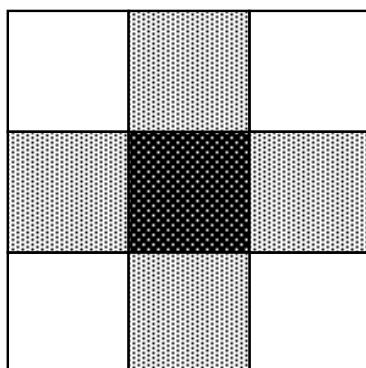


図 4.7: スモールダイヤモンド探索の探索点

をその点に移動する．図 4.7 に，スモールダイヤモンド探索の探索点を示す．また従来の TZSearch と同様に，探索中心から距離 1 が最良点であれば 2 点探索を行う．

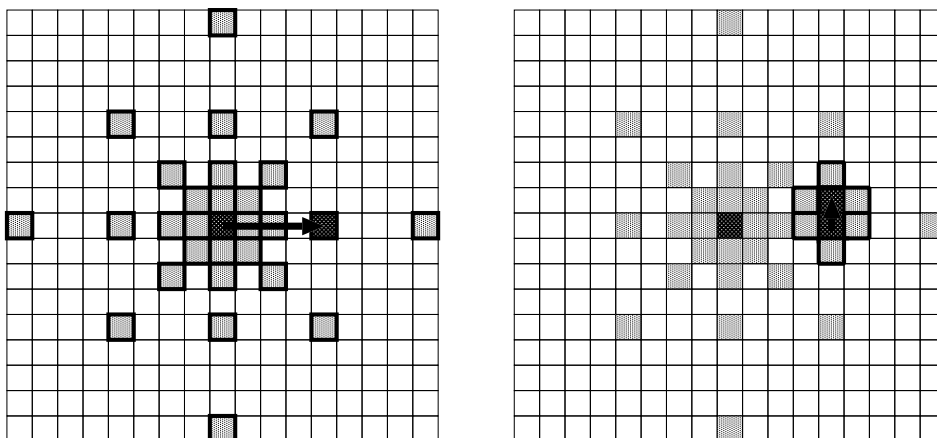


図 4.8: 切り替え型ダイヤモンド探索の例

図 4.8 では拡大型ダイヤモンド探索の後，スモールダイヤモンド探索を 2 回繰り返した場合の探索点と探索中心の推移を示している．スモールダ

イヤモンド探索で探索が終了すれば，無駄な拡大型ダイヤモンド探索を行う回数を減らすことができ，なおかつ予測ベクトルの近傍に動きが検出されるという探索特性を活かした近傍を中心に探索するアルゴリズムとなる．またスモールダイヤモンド探索で終了しない場合，再び拡大型ダイヤモンド探索を行うことで，SAD 最小のブロックが遠隔にある場合の誤検出を防ぐことができる．この手法のフローチャートを図 4.9 に示す．

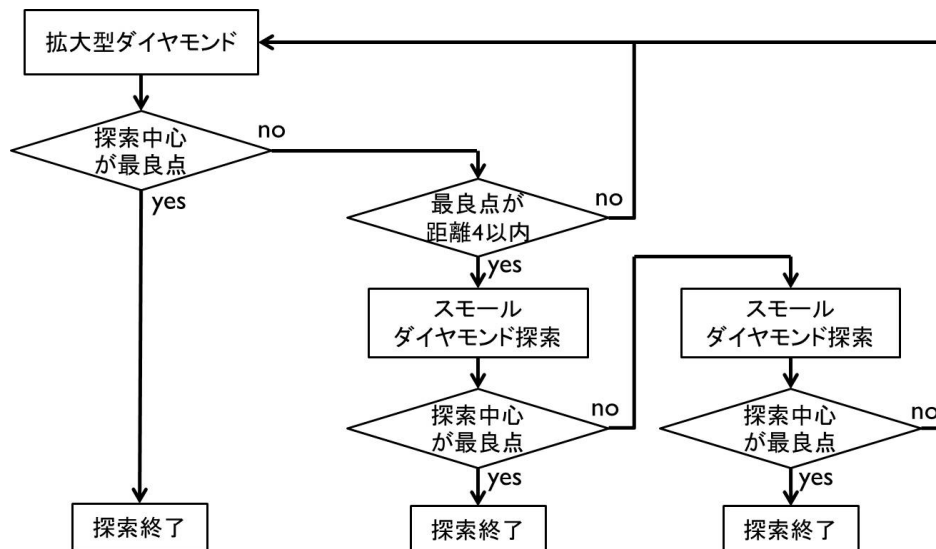


図 4.9: 切り替え型ダイヤモンド探索のフローチャート

4.3 ブロックサイズ適応型探索

4.3.1 ブロックサイズ 64x64 , 32x32

ブロックサイズ 64x64 と 32x32 とそれらを分割した形状のブロック (64x32 など) については，前述の切り替え型ダイヤモンド探索を用いる．この探

索法を用いることで、遠隔点も探索しつつより近傍を効率的に探索することができ、最小限の精度の低下に抑えつつ演算量を低減することを可能とする。

また、これらのブロックはサイズが大きく構成画素数も多いことから、動き検出で画素を間引いて処理しても探索精度への影響は小さい。更にはTZSearchでは前述した通り、予測ベクトルの精度が高く近傍で検出されることが多い。また、探索中心から離れた遠隔点はまばらになることから、近傍ほど高い精度で探索する必要はないため、よりサブサンプリングを粗くしても符号化効率などには影響しないと考えられる。このため64x64では探索中心から距離1から2画素精度、距離16から4画素精度、32x32では距離1から2画素精度、距離32から4画素精度でサブサンプリングを行う[5]。

このように距離が遠いほど間引き率を高めることで探索精度の低下を防ぎつつ演算量の低減を図っている。なお今回は簡単のため、縮小画像を生成せずにSAD演算において画素を間引いて計算を行った。図4.10、図4.11にこの方法を示す。

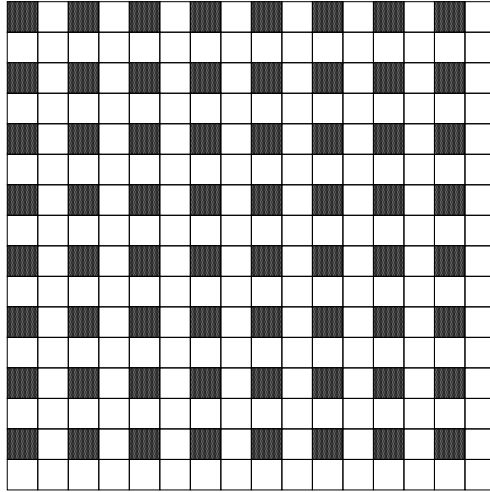


図 4.10: 2画素精度サブサンプリング

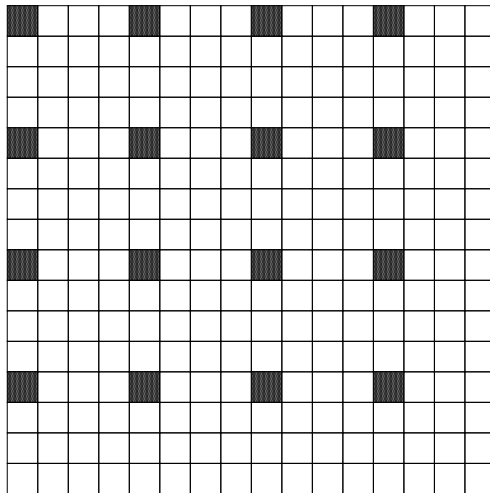


図 4.11: 4画素精度サブサンプリング

4.3.2 ブロックサイズ 16x16 , 8x8

ブロックサイズ 16x16 と 8x8 とそれらを分割した形状のブロックについては近傍をあらかじめ探索し、その最良点からスモールダイヤモンド探索を終了条件を満たすまで行う。16x16 については距離 8 周辺までをあらかじめ探索し (ただし予測ベクトルが水平、垂直方向ともに 0 なら距離 16)、8x8 では距離 2 までを探索する。それぞれの近傍点は図 4.12 に示す。

そしてこの近傍点の探索の中の最良点から、探索終了までスモールダイヤモンド探索を繰り返す。これは小さなブロックサイズでは、ほとんどの場合、近傍点の探索で動きベクトルが検出されることから、従来の拡大型ダイヤモンド探索を行う必要はないと考えられるためである。

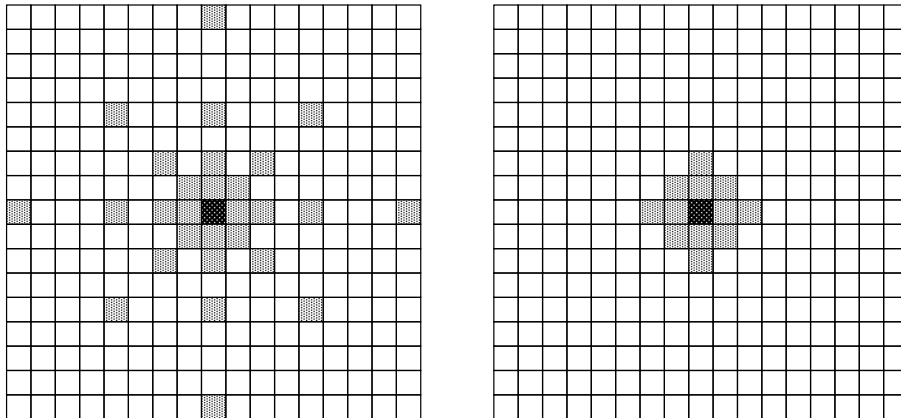


図 4.12: 16x16 と 8x8 の近傍探索点

またブロックサイズ 16x16 や 8x8 では、構成画素数が少なくなりすぎて

動き検出の精度が悪化してしまうためサブサンプリングは行わない。しかし小さなブロックでは大きなブロックと比べ、より予測ベクトルの近傍に動きが検出されるという特性があるため、拡大型ダイヤモンド探索を何度も繰り返さなくとも、十分な精度で動きを検出できる。また、最初の近傍探索での最良点が探索中心から距離 1 の場合、2 点探索を行った後にスモールダイヤモンド探索を行う。

4.4 全体のフローチャート

提案手法全体のフローチャートを示す。

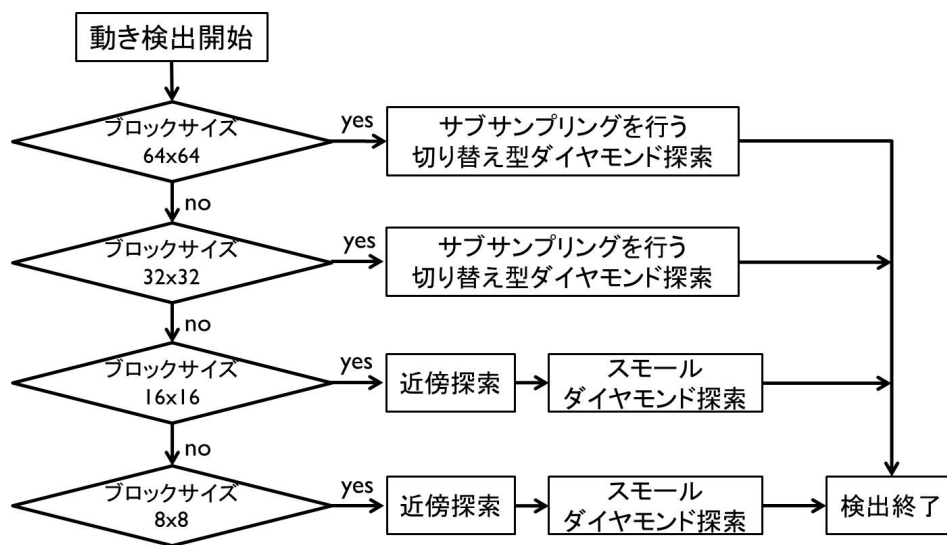


図 4.13: 全体のフローチャート

5 性能評価

5.1 評価

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て評価を行った．評価に用いたテストシーケンスは表 5.2 に示す．エンコード条件は表 5.3 に示す．

表 5.2: テストシーケンス

画像名	解像度	フレーム数	フレームレート
BasketballDrill	832x480	500	50
BQMall	832x480	600	60
Kimono	1920x1080	240	24
ParkScene	1920x1080	240	24

表 5.3: エンコード条件

符号化構造	ランダムアクセス GOP Size = 8 (IBBBBBBBBBBBBBB...)
探索範囲	± 64
CU サイズ/深さ	64/4
QP	32

評価はビットレート，YPSNR，AD(Absolute Difference) 演算回数，同一画質でのビットレート増加率である BD-rate の比較により行う．BD-rate はビットレートと YPSNR の関係をそれぞれ曲線で表し，積分によ

り曲線間の距離を求めた値である．デフォルトの TZSearch での評価結果を表 5.4，提案手法での評価結果を表 5.5，提案手法での BD-rate と SAD 演算回数低減率を 5.6 に示す．

表 5.4: TZSearch の結果

シーケンス	QP	ビットレート (kbps)	YPSNR(dB)	AD 演算回数 ($\times 10^{11}$)
BasketballDrill	22	3616.78	40.50	8.01
	27	1755.93	37.40	7.33
	32	865.62	34.46	6.53
	37	459.20	31.94	5.70
BQMall	22	3831.91	40.23	6.32
	27	1823.89	37.74	5.68
	32	931.35	35.02	5.06
	37	498.61	32.28	4.52
Kimono	22	4732.49	41.60	20.33
	27	2159.37	39.73	17.80
	32	1053.09	37.42	15.43
	37	533.29	35.03	13.35
ParkScene	22	7406.56	40.05	9.82
	27	3179.04	37.52	8.85
	32	1450.09	34.91	8.01
	37	670.63	32.39	7.25

表 5.5: 提案手法の結果

シーケンス	QP	ビットレート (kbps)	YPSNR(dB)	AD 演算回数 ($\times 10^{11}$)
BasketballDrill	22	3652.80	40.50	1.09
	27	1775.60	37.39	1.04
	32	873.70	34.45	0.98
	37	464.27	31.93	0.92
BQMall	22	3851.73	40.23	1.24
	27	1834.62	37.74	1.18
	32	935.90	35.01	1.13
	37	500.71	32.27	1.08
Kimono	22	4745.42	41.60	2.86
	27	2168.86	39.73	2.67
	32	1056.86	37.42	2.51
	37	534.51	35.03	2.37
ParkScene	22	7435.92	40.05	2.46
	27	3193.18	37.52	2.34
	32	1456.14	34.91	2.25
	37	672.92	32.38	2.17

表 5.6: 評価結果

シーケンス	QP	AD 演算回数低減率 (%)	BD-rate(%)
BasketballDrill	22	86.40	1.1
	27	85.85	
	32	85.02	
	37	83.82	
BQMall	22	80.43	0.7
	27	79.27	
	32	77.72	
	37	76.00	
Kimono	22	85.96	0.4
	27	85.01	
	32	83.76	
	37	82.23	
ParkScene	22	74.92	0.5
	27	73.55	
	32	71.95	
	37	70.04	

5.2 考察

提案手法は標準の TZSearch と比べて平均で 80.12% の AD 演算を低減し、BD-rate の悪化を最大でも BasketballDrill の 1.1% に抑えることができた。このことから提案手法では画質やビットレートの悪化がほとんどなく、演算量を低減できたことが分かる。

AD 演算回数低減率に差がでたことに関して、標準の TZSearch でこれらのシーケンスについて予測ベクトルとの誤差を調べたところ、BasketballDrill と Kimono では共に平均で約 90% が予測ベクトルとの距離 4 以内で動きが検出されるのに対して、BQMall と ParkScene では共に平均で約 95% が距離 4 以内で検出されていた。これにより予測が当たりにくいシーケンスでは演算量低減率が高いことが分かる。表 5.7 にそれぞれのテストシーケンスの予測誤差の割合を示す。ただし QP32 での結果のみを示す。

表 5.7: テストシーケンスの予測誤差

シーケンス	距離ごとの割合				
	0 - 4	5 - 8	9 - 16	17 - 32	33 -
BasketballDrill	90.24	2.33	1.92	1.85	3.66
BQMall	94.70	1.76	1.15	0.89	1.49
Kimono	91.09	3.13	1.92	1.48	2.38
ParkScene	97.03	1.07	0.61	0.48	0.81

また BD-rate が最も悪化した BasketballDrill の特徴としてはシーケン

ス全体で動きが不規則であるため、提案手法での予測ベクトルに対しての遠隔点における探索精度が落ちてしまったと考えられる。Kimono も全体として画面が動いているため予測ベクトルの近傍で検出される割合がBasketballDrill 同様に多少低かったものの、画面全体が同方向に進んでいるため探索は比較的行いやすく、精度の低下がみられなかったと考えられる。表 5.8 にそれぞれのテストシーケンスの動きベクトルの割合を示す。ただしQP32での結果のみを示す。これらから、必ずしも動きベクトルの大きさで動き検出の行いやすさが決まるのではなく、動きの不規則性により符号化効率に影響がでると考えられる。

表 5.8: テストシーケンスの動きベクトルの割合

シーケンス	距離ごとの割合						
	0 - 4	5 - 8	9 - 16	17 - 32	33 - 64	65 -128	129-
BasketballDrill	76.03	2.56	5.21	5.89	6.48	3.41	0.43
BQMall	41.39	16.06	20.11	13.08	6.87	2.10	0.38
Kimono	34.81	17.73	17.91	15.78	9.81	3.48	0.47
ParkScene	58.57	17.25	11.92	7.16	3.49	1.38	0.22

全体として、提案したアルゴリズムでは動きが不規則かつ速い場合に限り、ビットレートの増加と画質にわずかな劣化がみられる程度で、符号化効率低下を最小限に抑えることができた。このことから提案したアルゴリズムでは、どのようなシーケンスに対しても符号化効率を維持しつつ、大幅な演算量低減ができるといえる。

提案手法の改善点としては、予測が当たりにくいシーケンスで演算量低減率は高いものの、予測が当たりやすい動き検出が容易と考えられるシーケンスでは演算量低減率が低かったことから、予測が当たりやすい単純なシーケンスで更なる演算量低減が可能と考えられる。具体的な改善方法としては探索の早期打ち切りによる演算量低減法などが挙げられる。また悪化がみられた不規則なシーケンスに対応するために遠隔点の探索の効率化も改善点として挙げられる。

6 おわりに

本研究では近傍を効率的に探索する手法をブロックサイズ別に適用することで、画質とビットレートの悪化を最小限に抑え AD 演算回数を約 80% 低減できることを示した。また、提案手法の評価の過程で動きが複雑なシーケンスでわずかに BD-rate の悪化がみられることと、演算量低減の面で予測ベクトルが当たりやすい単純なシーケンスで更に演算量が低減できることが分かった。今後はこの結果に基づき、遠隔点に対するより効率的な探索と、予測ベクトルの精度に応じた早期打ち切りとを併せて組み込んでいくことで、演算量低減と高精度を両立するアルゴリズムの開発を目指す。

謝辞

本研究を行うにあたり，日頃から様々の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．また様々な面でお世話になった計算機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] 大久保榮ほか:H.265/HEVC 教科書，2013．
- [2] High Efficiency Video Coding(HEVC)，
<http://hevc.hhi.fraunhofer.de>
- [3] Xiu-li,Tang,Dai Sheng-kui,and Cai Can-hui. “ An analysis of TZSearch algorithm in JMVC. ” Green Circuits and Systems (ICGCS), 2010 International Conference on. IEEE, 2010.
- [4] Xiph.Org Foundation，<http://xiph.org/>
- [5] 立野，西村，近藤，佐々木，深澤，“ 非近傍点にサブサンプリング画像を用いる適応的拡大ダイヤモンド探索 ”，信学技報 (IE2014-49)，vol.114，no.233，pp.57-62，2014年10月．

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 .