

## 重み関数を用いた HEVC イントラ予測改良に関する検討 A Study on Enhancement of HEVC Intra Prediction Using Weight Function

松尾 翔平 Shohei Matsuo  
高村 誠之 Seishi Takamura  
清水 淳 Atsushi Shimizu

### 1. はじめに

近年画像の解像度やフレームレートが増大することにより、映像の有する情報量が飛躍的に増大している。映像の放送、通信(配信)、蓄積のために映像符号化技術は重要であり、映像符号化国際規格である H.264/AVC よりも高圧縮の規格が求められており、次世代高能率映像符号化 HEVC(High Efficiency Video Coding)の標準化が進められている。本稿では、HEVC におけるイントラ予測の方法と既存手法について述べ、その改良方法を提案する。また、評価実験を行い、本提案法の有効性を示す。

### 2. HEVC イントラ予測と既存改善手法

映像符号化ではブロックに分割して予測を行い、予測信号と原信号の差分信号に対して直交変換を施して、エントロピー符号化を行う。予測に関して、1つのフレーム内で処理対象ブロック周辺の復号信号から予測を行うイントラ予測と、復号済みの他フレームから動きベクトルを求めて予測信号を生成するインター予測がある。本章では HEVC のイントラ予測と既存改善手法を述べる。

#### 2.1 HEVC のイントラ予測処理

HEVC は H.264/AVC と同様にブロック単位で符号化を行うが、従来の  $16 \times 16$  のマクロブロックではなく、符号化ユニット(Coding Unit: CU)と呼ばれる単位で符号化を行う。CU は予測ユニット(Prediction Unit: PU)と変換ユニット(Transform Unit: TU)と呼ばれる予測及び変換ブロックに分割されて処理される。HEVC の参照ソフトウェアである HEVC test Model (HM)では、最大の CU は LCU(Largest CU)と呼ばれ、そのサイズは  $64 \times 64$  であり、最小の CU サイズは  $8 \times 8$  となっている。PU と TU はそれぞれ CU に対して独立に定義され、CU と同じサイズないしは四分木に分割される。PU については 1:3 の分割形状も許容する。従来のマクロブロックと比べて、大きいサイズが利用できるため、特に高解像度の画像にて圧縮効率が上がる利点がある。

HEVC のイントラ予測に関しては、H.264/AVC の予測モードが 9 モードあったのに対し、35 モードと増加している。具体的なモードの違いを表 1 にまとめる[1,2]。特に大きな差分となっているのが、Angular 予測モードであり、予測方向数が大きく増加している。その様子を図 1 に示す。

表 1 H.264/AVC と HEVC のイントラ予測の違い

予測モード	H.264/AVC	HEVC
DC	○	○(改良)
Angular	8 方向	33 方向
その他	Plane	Planar

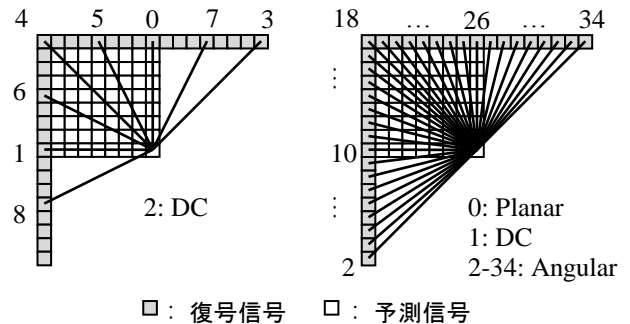


図 1 イントラ予測方向数(左: H.264/AVC, 右: HEVC)

図 1 に示される通り、HEVC では 33 方向と細かい予測方向をとることができる。柔軟な予測信号を生成することができる。表 1 に示した予測モードは、全 PU サイズ共通で用いられる仕様となっている。図 2 に示すように PU サイズが小さい場合は復号信号から予測信号までの予測距離が短いため、予測効率は十分高い形となるが、例えば  $16 \times 16$  のように PU サイズが大きい場合、その予測距離が長くなるため、予測方向先の予測信号の予測誤差エネルギーが増大する可能性が高くなるという問題点がある。

#### 2.2 イントラ予測符号化の既存改善手法

イントラ予測の符号化性能を向上させる方法として、(i) 予測信号の生成手段を工夫する手法[3-8]、(ii) 予測ブロックの分割形状を工夫する手法[9]、(iii) 予測方向の情報を利用して予測の次に入る変換処理を工夫する手法[10]などが存在する。

まず、(i) 予測信号の生成手段を工夫する手法について述べる。通常予測ブロックの復号参照信号は、予測対象ブロックの左上に存在するので外挿予測となるが、内部の符号化順序を変更して右及び下からの予測(内挿予測)を可能とする手法[3]や順次隣り合う画素を参照可能にして差分を計算して予測誤差エネルギーを低減する手法[4]が提案されている。また、著者らは、復号参照信号を予測対象ブロックの最近傍に位置する画素だけでなく、距離が離れた近傍画素も復号参照信号として利用する手法[5]と予測方向に沿って距離に応じて傾斜を付けて予測信号を生成する手法[6]、及びその組み合わせ手法[7]について提案している。HEVC では Angular 予測モードの方向数が増加したことにより、小数画素位置の復号参照信号も必要になったことから、補間の精度を上げて予測効率を向上させる手法[8]を提案した。

続いて、(ii) 予測ブロックの分割形状を工夫する手法としては、例えば  $8 \times 8$  の PU が選択された場合、 $2 \times 8$  ないしは  $8 \times 2$  のような 4 分割を行って細長い矩形領域に分割して、復号参照信号と予測対象信号の距離を短くすることで予測精度を向上させる方法[9]が提案されている。

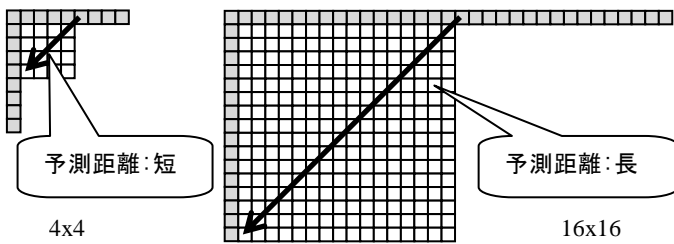


図2 PUサイズと予測距離の関係

最後に(iii)後段の変換処理を工夫する手法としては、イントラ予測モードの予測方向に応じて、変換に用いる基底を変更する方法[10]が挙げられる。

以上、イントラ予測符号化の既存改善手法を述べたが、HEVCの場合より大きなブロックサイズを許容し、かつ予測方向数が増加するという点が H.264/AVC とは異なる。前記の点を考慮して、予測信号の生成方法には検討の余地があると思われる。本稿では、2.1 にて記載したイントラ予測の問題点を解決するために、柔軟に予測信号を生成できる方法を提案する。

### 3. 提案イントラ予測手法

提案するイントラ予測方法を説明する。まず、本提案手法の着眼点と概念について述べ、詳細な符号化アルゴリズムを述べる。

#### 3.1 着眼点

Angular 予測モードは、表 1 にて示した通り、予測モードの中で数が多い予測モードであり、Angular 予測モードの改善はイントラ予測の符号化効率向上に結び付きやすいと考えられる。また、HEVC では前章にて述べたとおり、詳細な予測方向を有し、かつ大きな予測ブロックサイズが選択できる。図 2 に示す通り、予測ブロックサイズが大きくなると、特に Angular 予測モードでは予測距離が非常に長くなるため、図 2 を例にとると、ブロックの左下部分における予測精度が低下することが懸念される。予測方向の指し示す先の復号参照信号が図 2 では存在しているため、予測距離の短い復号参照信号を予測に用いた方が予測精度の改善が狙えると言える。

予測モード 10(水平予測)から 26(垂直予測)では、予測方向の指し示す先に復号参照信号が存在しないことになるが、予測モード 2 から 9 及び 27 から 34 では、予測方向先に復号信号が存在するため、逆方向の Angular 予測信号を生成することができる。本提案法では、順方向の予測信号に対して逆方向の予測信号の重み付き和を取ることで、予測誤差エネルギーを低減して符号化効率を向上させる。提案法の概念図を図 3 に示す。

#### 3.2 符号化アルゴリズム

順方向の Angular 予測信号と逆方向 Angular 予測信号を合成して予測信号を生成する詳細な符号化アルゴリズムを以下に示す。本提案手法は 2 種類のアプローチからなる。

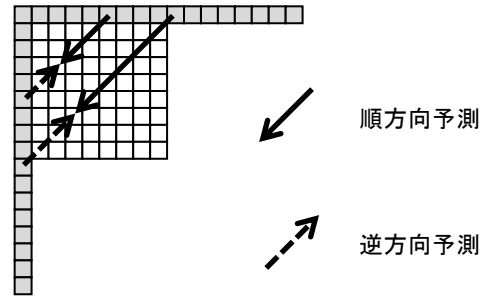


図3 提案法の概念図

[提案手法 A: ブロック適応切り替え機構なし]

**Step A-1:** 提案イントラ予測モードの集合を  $\phi$  とし、予測モード 2 から 9 及び 27 から 34 にセットする。

**Step A-2:** イントラ予測モードをセットする。当該モードが  $\phi$  に含まれるならば、**Step A-3** に移行する。含まれないならば、従来の HEVC イントラ予測[2]と同じ処理を適用して予測信号を生成して符号化を実施する。その後、**Step A-4** に移行する。

**Step A-3:** 重み付き予測信号を生成する。順方向の Angular 予測信号を  $P1[i][j]$  ( $i, j$  はブロック内の位置)、逆方向の Angular 予測信号を  $P2[i][j]$  とする。このとき、重み付き予測信号を  $P[i][j]$  とすると、

$$P[i][j] = f(d) \times P1[i][j] + (1 - f(d)) \times P2[i][j] \quad (1)$$

で生成する。 $f(d)$  は重み関数であり、 $d$  は予測方向に沿ったときに復号参照信号を起点 ( $d=0$ ) および予測方向先にある復号参照信号を終点 ( $d=1$ ) とし、その復号参照信号から予測対象信号までの相対的距離を意味する。重み関数の事例を図 4 に示す。前記(1)より求めた重み付き予測信号にて符号化を実施する。

**Step A-4:** 次の予測ブロックへ処理を移行する。全ての処理ブロックが終わるまで **Step A-1** から **A-4** を繰り返す。

本提案法は輝度信号だけでなく、色差信号についても同様に適用できる。

[提案手法 B: ブロック適応切り替え機構あり]

**Step B-1:** 提案イントラ予測モードの集合を  $\phi$  とし、予測モード 2 から 9 及び 27 から 34 にセットする。

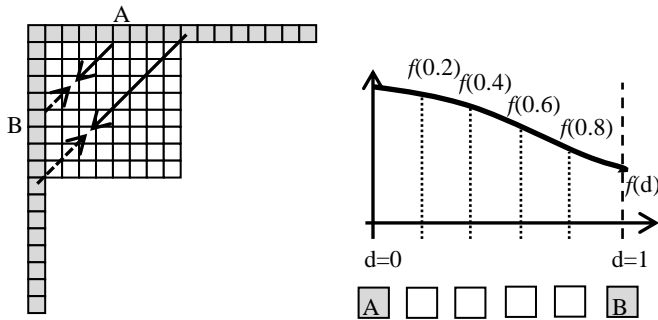
**Step B-2:** イントラ予測モードをセットして、従来の HEVC イントラ予測[2]と同じ処理を適用して予測信号を生成する。この順方向予測の予測信号を保存する。選択した当該モードが  $\phi$  に含まれるならば、**Step B-3** に移行する。含まれないならば、**Step B-4** に移行する。

**Step B-3:** 重み付き予測信号を生成する。生成する方法は **Step A-3** にて示されている方法と同じである。

**Step B-4:** **Step 2** で生成した順方向予測の予測信号と、**Step 3** で生成した重み付き予測信号とを用いて、それぞれの場合での RD コストとを計算し、RD コストが良い方を選択する。その予測信号を最終的な予測信号として採用して、本符号化に移る。このとき、重み付きか否かの情報をオーバーヘッドとして符号化し、復号側へ伝送する。

**Step B-5:** 次の予測ブロックへ処理を移行する。全ての処理ブロックが終わるまで **Step B-1** から **B-5** を繰り返す。

本提案法は輝度信号だけでなく、色差信号についても同様に適用できる。

図4 重み関数の例 ( $f(d)=\exp(-d^2)$ )

### 3.3 既存改善手法との比較

本提案手法と 2.2 にて記載したイントラ予測の既存改善手法との比較を述べる。本提案手法は(i)の予測信号生成手段を工夫する方法に属する。内挿予測方法[3]と傾斜付与方法[6]が概念的に近いが、本提案手法の重み関数のように柔軟に重みを設定することができない。重み関数の設定次第で、予測誤差エネルギーをより低減でき得る。また、残りの既存改善手法[4-5,7-10]に関しては、本提案手法との併用が可能なので、同時に適用することで大きな符号化効率改善を期待できる。

## 4. 評価実験

### 4.1 実験条件

本提案手法を HEVC 拡張規格用参照ソフトウェア HM9.0 RExt (Range Extensions) [11]に実装した。符号化実験は、標準化共通条件[12]に従い、RGB4:4:4の原画像8種、及びYUV4:4:4/4:2:2の原画像をそれぞれ7種ずつの計22種類の画像を用いた。解像度は1920×1080及び2560×1600の2種類で、ビット深度は8/10/12ビットの3種類である。各画像の10秒分(240~600枚)のフレームをQP={17, 22, 27, 32, 37}の5種類のQPを用いて全てイントラ符号化した。

提案手法の重み付き予測信号を生成する場合の重み関数は  $f(d)=\exp(-d^2)$  を用いて、提案手法 A/B 共通で同じ関数を利用した。また、輝度色差両方向同じ関数を設定し、全ブロックサイズに対して本提案方法を適用した。

### 4.2 結果と考察

重み付き予測信号を全て場合で適用して、ブロック単位で提案手法の on/off の切り替えを行わない提案手法 A の結果を表 2 及び 3 に示す。QP={22, 27, 32, 37}の4点で計算した中ビットレート(標準化において Main-tier と呼ばれる領域)での結果が表 2 であり、QP={17, 22, 27, 32}の4点で計算した高ビットレート(標準化において High-tier と呼ばれる領域)での結果が表 3 となる。本実験におけるアンカーデータは、提案手法を導入していないオリジナルの HM9.0 RExt での結果である。

表 2 及び 3 より、符号化性能の改善率は輝度でアンカー比約 0.3% であり、色差の 2 チャンネルについては若干輝度よりも低かった。高ビットレートの符号化利得は、中ビットレートと比較して平均的に低い値であった。高レートでは復号信号が原信号に近い形で復元されるため、

表 2 提案手法 A の符号化性能と処理時間  
(QP={22, 27, 32, 37}, Main-tier)

	Y	U	V
RGB4:4:4	-0.32	-0.31	-0.28
YCbCr4:4:4	-0.38	-0.27	-0.25
YCbCr4:2:2	-0.24	-0.07	-0.01
Overall [%]	-0.31	-0.22	-0.19
Enc-time [%]	103.9		
Dec-time [%]	104.2		

表 3 提案手法 A の符号化性能と処理時間  
(QP={17, 22, 27, 32}, High-tier)

	Y	U	V
RGB4:4:4	-0.24	-0.23	-0.20
YCbCr4:4:4	-0.32	-0.20	-0.18
YCbCr4:2:2	-0.20	-0.09	-0.06
Overall [%]	-0.25	-0.17	-0.15
Enc-time [%]	103.2		
Dec-time [%]	103.9		

予測効率がよく、本提案法の利得が小さくなるのが原因と考えられる。ただし、色差の結果をみると、YCbCr4:2:2 の結果では高レートでの符号化利得が中ビットレートよりも大きい。処理時間に関しては、表 2 及び 3 ともに同じ傾向であり、符号化時間の増加率は約 3.5%、復号処理時間の増加率は約 4% であった。

続いて、ブロック単位で重み付き予測を用いるかどうかを RD コストから判定して、ブロック単位で on/off を切り替える提案手法 B の結果を表 4 から 7 に示す。LCU 単位で切り替えた結果が表 4 及び 5 であり、PU 単位で切り替えた結果が表 6 及び 7 である。

表 4 より、符号化性能の改善率は輝度でアンカー比約 0.6% であり、色差の 2 チャンネルについてもほぼ同様の符号化利得が得られた。高ビットレートとなる表 5 の結果では符号化利得が約 0.38~0.45% と低下していた。これは提案手法 A の場合と同様に、復号信号が原信号により近いこと、予測効率がよく、本提案法の利得が小さくなるのが原因と考えられる。どちらのビットレート帯域においても、提案手法 A と比べて良い利得が得られていた。LCU 単位で適応的に重み付き予測を on/off することにより、利得が増大したと考えられる。本実験では LCU を 64×64 に設定しており、オーバーヘッドの符号量も少なく済むため、良好な結果が得られた。

処理時間については、LCU 単位で従来の予測信号を用いるパターンと本提案法の重み付き予測信号を用いるパターンを 2 つとも試行して良い方を選択する処理になっているため、大幅に符号化時間が増大している結果となっていた。メモリを低減する理由から本実験では前記 2 回の試行後に 3 回目の本符号化を実施するため、約 3 倍近い処理量となっていた。実装を工夫することで約 2 倍程度に収めることができると考えられる。また、復号時間については、符号化側で生成したオーバーヘッド情報を復号して重み付き予測信号を生成するか否かが判断できるため、符号化処理ほど時間が増大していない結果となっていた。重み付き信号を生成する処理のみが増加分となり、提案手法 A と同様、復号時間は約 3.5~4% の増加であった。

表4 提案手法Bの符号化性能と処理時間  
(QP={22, 27, 32, 37}, Main-tier, LCU単位での on/off)

	Y	U	V
RGB4:4:4	-0.58	-0.60	-0.51
YCbCr4:4:4	-0.71	-0.69	-0.81
YCbCr4:2:2	-0.53	-0.42	-0.46
Overall [%]	-0.60	-0.57	-0.59
Enc-time [%]	305.4		
Dec-time [%]	103.9		

表5 提案手法Bの符号化性能と処理時間  
(QP={17, 22, 27, 32}, High-tier, LCU単位での on/off)

	Y	U	V
RGB4:4:4	-0.40	-0.41	-0.35
YCbCr4:4:4	-0.55	-0.46	-0.53
YCbCr4:2:2	-0.40	-0.27	-0.30
Overall [%]	-0.45	-0.38	-0.39
Enc-time [%]	303.6		
Dec-time [%]	103.5		

表6及び7の結果は、PU単位で重み付き予測を切り替えた結果である。LCU単位で切り替えた場合と比較して、符号化利得は全体的に小さくなっていった。これはオーバーヘッドの符号量が大幅に増え、予測誤差エネルギー低減の効果が相殺されたことが原因と考えられる。

PU単位で切り替え、かつオーバーヘッド情報を伝送しない場合の理想的な利得を表8に示す。復号ができないため、処理時間は省いた。表8が示す通り、輝度にて約1%程度改善した。表6との差がオーバーヘッドにて効果が相殺された分となる。PU単位で切り替える場合、オーバーヘッド情報の削減が重要な課題と思われる。

## 5. まとめ

本稿では、HEVCのイントラ予測と問題点について論述し、予測方向先に復号参照信号が存在するAngular予測モードにて、順方向の予測信号と逆方向の予測信号の重み付き信号を生成することで、予測誤差エネルギーの増大を抑制する改善方法を検討した。実験の結果、LCU単位で提案手法を切り替えることで約0.45~0.6%の符号化効率改善を達成した。今後は、符号化処理量削減の検討とオーバーヘッド情報の削減方法の検討を行う。また、本稿の実験で用いた重み関数は、最適であるかどうかは保証されていない。入力画像の解像度や特性、符号化条件、ビットレートによって最適な重み関数は異なると考えられるので、重み関数を変更した場合の符号化性能の変動を確認する予定である。また、ブロックサイズや予測モードに応じて、重み関数を切り替えるようにすれば、より高い符号化利得が得られる可能性もあると考えられるので、ブロックサイズや予測モードに応じた重み関数切り替え機構の検討を実施する。

## 謝辞

本検討に用いた画像の一部は、NTTドコモ様の許諾を得て使用したものです。この場を借りてお礼申し上げます。

表6 提案手法Bの符号化性能と処理時間  
(QP={22, 27, 32, 37}, Main-tier, PU単位での on/off)

	Y	U	V
RGB4:4:4	-0.36	-0.16	-0.09
YCbCr4:4:4	0.01	-0.05	-0.11
YCbCr4:2:2	0.12	0.14	0.15
Overall [%]	-0.09	-0.03	-0.02
Enc-time [%]	171.2		
Dec-time [%]	102.7		

表7 提案手法Bの符号化性能と処理時間  
(QP={17, 22, 27, 32}, High-tier, PU単位での on/off)

	Y	U	V
RGB4:4:4	-0.33	-0.15	-0.09
YCbCr4:4:4	-0.16	-0.02	-0.04
YCbCr4:2:2	-0.06	0.11	0.13
Overall [%]	-0.19	-0.03	-0.00
Enc-time [%]	172.2		
Dec-time [%]	102.4		

表8 提案手法Bの理想符号化性能(オーバーヘッドなし)  
(QP={22, 27, 32, 37}, Main-tier, PU単位での on/off)

	Y	U	V
RGB4:4:4	-1.00	-0.79	-0.68
YCbCr4:4:4	-1.10	-0.82	-0.91
YCbCr4:2:2	-0.91	-0.38	-0.39
Overall [%]	-1.01	-0.67	-0.66

## 参考文献

- [1] "Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding," ISO/IEC 14496-10, 2005.
- [2] B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, Y.-K. Wang and T. Wiegand, "High Efficiency Video Coding (HEVC) text specification draft 10 (for FDIS & Last Call)," JCTVC-L1003\_v34, 12th meeting of JCT-VC, January, 2013.
- [3] 塩寺, 谷沢, 中條, "ブロックベース外挿/内挿予測に基づくイントラ符号化," 画像符号化シンポジウム PCSJ2006, pp. 123-124, November, 2006.
- [4] 高橋, 村上, "原画像を利用した新しい画面内予測によるH.264/AVC拡張方式," 画像符号化シンポジウム PCSJ2007, pp. 43-44, October, 2007.
- [5] 松尾, 坂東, 高村, 如澤, "複数参照ラインを用いたイントラ予測改善方法," 電子情報通信学会論文誌D, Vol. J94-D, No. 12, pp. 1940-1942, December, 2011.
- [6] 松尾, 高村, 上倉, 八島, "傾斜付きイントラ予測の基礎検討," 画像符号化シンポジウム PCSJ2007, October, 2007.
- [7] S. Matsuo, S. Takamura, and Y. Yashima, "Intra prediction with spatial gradients and multiple reference lines," Proc. of PCS2009, May, 2009.
- [8] S. Matsuo, S. Takamura, and H. Jozawa, "Improved intra angular prediction by DCT-based interpolation filter," Proc. of EUSIPCO 2012, pp. 1568-1572, August, 2012.
- [9] X. Cao, C. Lai, Y. Wang, and Y. He, "Short distance intra coding scheme for HEVC," Proc. of PCS2012, pp. 501-504, May, 2012.
- [10] Y. Yen, and M. Karczewicz, "Improved intra coding," ITU-T Q.6/SG16 VCEG, VCEG-AG11, October, 2007.
- [11] HM RExt software 9.0, <https://hevc.hhi.fraunhofer.de/trac/hevc/browser/branches/HM-range-extensions>.
- [12] D. Flynn, "Common test conditions and software reference configurations for HEVC range extensions," JCTVC-K1006, 11th JCT-VC meeting, Shanghai, CN, 10-19 October, 2012.