

RI-003

スパース性制約に基づく適応イントラ予測器の最適設計 Optimal design of adaptive intra predictors based on sparsity constraint

佐山雄一[†], 坂東幸浩[†], 高村誠之[†], 清水淳[†]Yuichi SAYAMA[†], Yukihiro BANDO[†], Seishi TAKAMURA[†], and Atsushi SHIMIZU[†]

1 はじめに

高効率な映像符号化を実現するには、イントラ予測の予測誤差低減は重要な課題となる。映像符号化の国際規格では近傍画素間の相関を利用したイントラ予測が検討されてきた。しかし、これまでのイントラ予測では、予測係数が予め固定されているため、画像によっては予測誤差を十分に低減できないことが問題として報告されている。例えば、HEVC[1]の方向性予測は、画素値変化の方向が直線的である仮定に基づき設計されている。従って、画素値変化の方向が変化する場合、予測誤差が増大する。

上述の問題に対する HEVC イントラ予測に対する改善策は、(1) 参照画素として近傍画素を用いる方法 [2, 3, 4], (2) 参照画素として非近傍画素を用いる方法 [5, 6, 7, 8, 9] に分類できる。まず、参照画素として近傍画素を用いる方法について概説する。文献 [2] では、予測の構造が双方向予測に拡張され、さらに、参照画素からの距離に基づく重み付けがなされている。文献 [3] では、方向性予測の方向の粒度が 65 通りに精緻化されている。文献 [4] では、画素値予測の方向が円弧により表現され、曲線的な予測に拡張されている。

しかしながら、これらの改善策は発見的であり、画素値予測の方向に関して、予め定められたモデルが固定的に用いられている。このため、方式改善の充分性は検証されておらず、未知の画像に対する予測性能は、保証の限りではない。次に、参照画素として非近傍画素を用いる方法について概説する。この方法は参照領域の位置の同定に必要な情報を符号化情報として追加する方法 [5, 6, 7, 8] と、追加しない方法 [9] に大別される。前者は、符号化時に参照領域を探索し、探索結果を付加情報として符号化情報に追加する。後者は、符号化器/復号器で探索のルールを共有し、参照領域の位置に関する符号化情報を省略している。これらは、画像内の自己相似性を予測処理に利用したアプローチである。しかし、画像内の自己相似性の程度にはばらつきが大きく、自己相似性が低い画像に対しては、予測誤差の低減は望めない。

そこで本稿では、従来の近傍画素を参照するイントラ予測を包括的に表現可能な予測構造を導入し、画像毎に予測器を設計する。一方、予測器を表現する予測係数が付加

情報となるため、予測係数の発生符号量の抑制が課題となる。そこで予測係数の符号量低減と予測誤差の符号量低減を両立すべく、予測係数に対して、スパース性に関する制約条件を付与し、同条件下で予測誤差を最小化する予測器設計問題を定式化する。予測器設計問題の求解法として、LASSO[10] 求解法の 1 つである LARS[11] をマルチクラス の LASSO へ対応可能な形に拡張した方法を提案する。

2 信号特性に適応する線形イントラ予測方式

2.1 提案するイントラ予測の定式化

提案手法では 1 枚の画像を矩形ブロックに分割し、矩形ブロックごとに予測を行う。以降では、予測処理の単位となる矩形ブロックを予測ブロックと呼ぶ。画素値の予測は各予測ブロックの直左もしくは直上の参照画素値を用いる。予測ブロックと参照画素の位置関係の例を図 1 に示す。図 1 の例では、予測ブロックを構成する 4×4 個の画素値を予測するため、予測ブロックの直左もしくは直上の計 13 個の画素を参照画素とする。

画像内の局所的な特性変化に対応するため、予測ブロックに対するクラス分類を行い、クラス毎に予測器を設計する。予測ブロック内の画素値と原画像の画素値の誤差である予測誤差は低く抑えることが望ましい。一方で、同一画像内であっても予測ブロックはそれぞれ信号特性が異なる。そこで、類似した信号特性を持つ予測ブロックを 1 つのクラスにまとめ、各クラスに対して予測器を設計する。予測ブロックに対するクラス分類の例を図 2 に示す。図 2 の例では 3×5 個の予測ブロックを 3 種類のクラスに分類している。

図 1 に示すような予測ブロックに対する近傍画素を用いた線形予測は次式の形に一般化できる。

$$\hat{y}_c^i = A_c x_c^i. \quad (1)$$

ここで、 i は予測ブロックの番号、 $c \in \{1, \dots, C\}$ はクラス番号である。 $x_c^i = (x_c^i[1], \dots, x_c^i[l], \dots, x_c^i[M])^T$ は参照画素を要素とする M 次元の列ベクトルである。 M は予測に用いる参照画素数である。 $\hat{y}_c^i = (\hat{y}_c^i[1], \dots, \hat{y}_c^i[l], \dots, \hat{y}_c^i[N])^T$ は予測ブロック内の画素値を要素とする N 次元の列ベクトルである。 N は予測ブロックの画素数である。 j は予測ブ

[†] 日本電信電話株式会社 NTT メディアインテリジェンス研究所

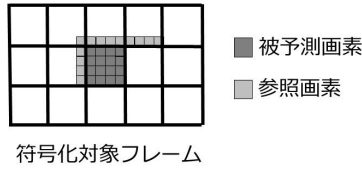


図 1: 予測ブロックと参照画素の例

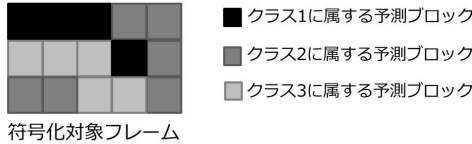


図 2: 予測ブロックのクラス分類の例 (3 クラスに分類した例)

ロック内の画素位置を示す指数である。 A_c はクラス c における予測係数を格納した行列 (N 行, M 列) である。 $a_c^{(j)}$ を予測係数行列 A_c の j 行目のベクトルとすると、 $a_c^{(j)}$ はクラス c に属する予測ブロックの j 番目の画素値を予測するための係数である。以降では、予測係数行列 A_c の各行ベクトル $a_c^{(j)}$ を単画素予測係数ベクトルと呼ぶ。単画素予測係数ベクトル $a_c^{(j)}$ および参照画素 x_c^i を用いてクラス c に属する i 番目の予測ブロック内の j 番目の画素値は $a_c^{(j)} x_c^i$ と予測する。

式 (1) は HEVC の参照画素平滑化、境界予測値修正、Planar 予測を含むすべてのイントラ予測を包含している。HEVC イントラ予測ではイントラ予測モード番号毎に参照画素位置が固定されており、参照画素値に乗ずる予測係数も予め定数として設定されている。例えば、 4×4 画素の PU に対する垂直方向予測 (モード番号 26) の場合、図 3 における黒太線の矩形で示される被予測画素に対し、白太線の矩形で示される直上の参照画素値を予測画素値として定める。これに対し、式 (1) の予測モデルでは参照画素位置、予測係数も適応的に設定可能である。従って、HEVC におけるイントラ予測よりも高い予測誤差低減が期待される。

提案手法で設定すべき情報は、各予測ブロックへ与えるクラス番号と、各クラスに与える予測係数行列 A_c の 2 種類である。以降の節では、クラス番号および予測係数行列の最適化方法について述べる。

2.2 クラス番号と予測係数行列の交互最適化

クラス番号と予測係数行列を決定するため、クラス番号と予測係数行列の最適化を交互に行う。最適なクラス番号および予測係数行列を同時に求めるのは困難である。そこで、一方の情報を固定して他方を最適化し、準最適解を求める。上記最適化は、発生符号量が収束するまで交互に繰

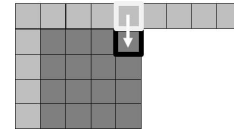


図 3: HEVC の垂直方向予測における被予測画素と参照画素の例

り返される。

クラス番号の最適化では、予測係数を固定して RD 最適化に基づきクラス番号を決定する。各予測ブロックは予測ブロックの符号化で生じる RD コストが最も小さくなるクラスを選択する。クラス番号最適化の際には予測係数は固定されるため、各予測ブロックは RD 最適化のための符号化を C 回 (C はクラス数) 行うことでクラスを選択できる。

予測係数の最適化では、クラス番号を固定して、予測係数行列内の零値要素数の比率に上限を与えた制約条件下で予測誤差を最小化する予測係数行列を求める。予測係数行列は付加情報となるため、その符号量の抑制が課題となる。そこで、予測誤差の低減と予測係数行列の符号量低減を両立するため、予測係数行列にスパース性の制約条件を課し、最適化する。以降の節では、スパース性を課した予測係数行列の最適化方法について述べる。

2.3 予測係数行列設計問題の定式化

クラス番号を固定して、スパース性に関する制約条件下で予測誤差を最小化するため、次式で示す L1 ノルムに関する制約付き二乗誤差最小化問題を考える。

$$\begin{aligned} \min_{A_1, \dots, A_C} & \sum_{j=1}^N \sum_{c=1}^C \sum_{i \in S_c} (y_c^i[j] - a_c^{(j)} x_c^i)^2, \\ \text{subject to} & \|a_c^{(j)}\|_1 \leq \theta_{c,j}, \sum_{c=1}^C \sum_{j=1}^N \theta_{c,j} = \bar{\theta}. \end{aligned} \quad (2)$$

ここで、 S_c はクラス番号 c が与えられた予測ブロック番号の集合である。 $\|a_c^{(j)}\|_1$ は、単画素予測係数ベクトル $a_c^{(j)}$ における予測係数の絶対値の総和、すなわち L1 ノルムである。 $\theta_{c,j}$ は各単画素予測係数ベクトルに対して定める係数の絶対値和の上限である。以降では、上記 $\theta_{c,j}$ を L1 ノルム上限と呼ぶ。 $\bar{\theta}$ は、全クラスにおける予測係数の絶対値和の上限である。以降では、 $\bar{\theta}$ を累積 L1 ノルム上限と呼ぶ。予測係数の絶対値和に関する制約条件を付与することで、経験的にスパース性を有する予測係数が得られる事が知られている [10]。

c, j を固定した単体の単画素予測係数ベクトル $a_c^{(j)}$ 設計問題は LASSO [10] として知られる問題と同一形式であり、次式で示される。

$$\begin{aligned} \arg \min_{a_c^{(j)}} & \sum_{i \in S_c} (y_c^i[j] - a_c^{(j)} x_c^i)^2, \\ \text{subject to} & \|a_c^{(j)}\|_1 \leq \theta_{c,j}. \end{aligned} \quad (3)$$

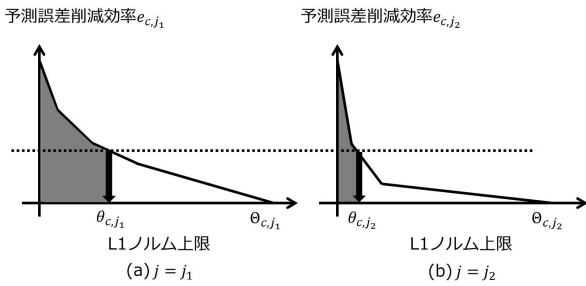


図 4: 二つの画素位置 $j = j_1, j = j_2$ における予測誤差削減効率 $e_{c,j}$, L1 ノルム上限 $\theta_{c,j}$, 予測誤差和の関係の例

以降では, LASSO の求解法について概説する. LARS[11] は L1 ノルム上限の値域を複数の部分区間に分割し, 部分区間毎に LASSO の最適解を求める方法である. Shooting Algorithm[12], FISTA[13] は目的関数や制約関数の凸性を利用して, 指定された L1 ノルム上限に対する最適解を高速に求める方法である.

しかし, 既存の LASSO 求解法では式 (2) で示される単画素予測係数ベクトル設計問題の求解は不可能である. なぜなら, 式 (2) において, 累積 L1 ノルム上限が定められた際に与えるべき L1 ノルム上限を定める方法がないためである. 式 (2) の最適化のためには, 同式における $N \times C$ 個の L1 ノルム上限を適切に設定可能な求解法が必要である.

2.4 予測係数行列設計問題の最適化

式 (2) の最適化問題に対して, 前述の LARS を拡張した求解法を導出する. まず, 求解の準備として, 各単画素予測係数ベクトルに対して, 単画素予測係数ベクトル $a_c^{(j)}$ に対する予測誤差削減効率 $e_{c,j}$ を次式として定義する.

$$e_{c,j} = \lim_{\delta_\theta \rightarrow 0} \frac{f_c^{(j)}(\theta_{c,j}) - f_c^{(j)}(\theta_{c,j} + \delta_\theta)}{\delta_\theta}. \quad (4)$$

ここで, δ_θ は, 現在の L1 ノルム上限における更新量である. $f_c^{(j)}(\theta)$ はクラス c に属する予測ブロックにおける j 番目の被予測画素に関して, 与えられた L1 ノルム上限 θ に対して式 (3) を求解して得られる単画素予測係数ベクトル $a_c^{(j)}$ で実現可能な予測誤差の総和を返す関数であり, 次式で示される.

$$f_c^{(j)}(\theta) = \begin{cases} \min_{a^{(j)}} & \sum_{i \in S_c} (y_c^i[j] - a_c^{(j)} x_c^i)^2, \\ \text{subject to} & \|a_c^{(j)}\|_1 \leq \theta. \end{cases}$$

提案する求解法では, L1 ノルム上限と予測誤差削減効率の関係に着目する. 単画素予測係数ベクトル $a_c^{(j)}$ に対して, 予測誤差の二乗和を最小化する設定がなされたベクトルの L1 ノルムを $\Theta_{c,j}$ と定義し, 最小二乗誤差 L1 ノルムと呼ぶ. L1 ノルム上限 $\theta_{c,j}$ は最小二乗誤差 L1 ノルム $\Theta_{c,j}$

を超えない範囲において, 最適な単画素予測係数ベクトルと一対一に対応する. 式 (4) に従い, 単画素予測係数ベクトルに対して, 予測誤差削減効率が定まる. 従って, 予測誤差削減効率に対して L1 ノルム上限が一意に定まる.

予測誤差削減効率に基づき最適な L1 ノルム上限を設定する考え方を図 4 にて説明する. 同図では, L1 ノルム上限の設定対象として, あるクラス c について, 2 つの単画素予測係数ベクトル設計問題が与えられた場合を考える. 各単画素予測係数ベクトル設計問題において, L1 ノルム上限と予測誤差削減効率が同図に示される関係にあったとする. このとき, 同図において, 横軸は L1 ノルム上限であり, 縦軸は L1 ノルム上限を定めた場合の予測誤差削減効率である. L1 ノルム上限は, 予測器において使用可能な予測係数の総量 (L1 ノルムで評価) の上限である. 予測誤差削減効率は, 使用する予測係数の総量の増加に伴う予測画像の電力の増分 (画素値の二乗和で評価) と解釈することもできる. 同図の灰色領域の面積は, 各々 L1 ノルム上限を $\theta_{c,j_1}, \theta_{c,j_2}$ としたときの予測画像の画素値の二乗和を表す. L1 ノルム上限を最小二乗誤差 L1 ノルムに設定すると, 予測誤差の二乗和を最小化する単画素予測係数ベクトルが得られる. 従って, 白色領域の面積は, L1 ノルム上限の増加により削減可能な予測誤差の二乗和を示す.

式 (2) で示される最適化問題は, 同図において, 2 つの L1 ノルム上限の和を一定に保つ制約を課し, 灰色領域の面積を最大化する問題と等しい. その理由は以下の通りである. 式 (2) における目的関数は予測誤差の二乗和の最小化である. これは, 線形予測で達成可能な予測誤差の二乗和と現在の L1 ノルム上限の設定で達成される予測誤差の差分を表す白色領域の最小化と等価である. 従って, 灰色領域の面積を最大化する問題と等価である. また, 式 (2) における制約関数は L1 ノルム上限の総和が一定である制約を意味する.

図 4 の灰色領域の面積和を最大化するために, 2 つの単画素予測係数ベクトル問題における予測誤差削減効率が等しくなるように L1 ノルム上限 $\theta_{c,j_1}, \theta_{c,j_2}$ を設定すればよい. その理由は以下の通りである. まず, 予測誤差削減効率は以下の性質を持つ.

- L1 ノルム上限の増加により, 予測誤差は減少する. 図 4 では, 予測誤差削減効率が非負であることと対応する.
- L1 ノルム上限の増加により, 予測誤差削減効率は減少する. 図 4 では, 予測誤差削減効率が単調減少であることと対応する.

このとき, ある c について, 仮に $e_{c,j_1} < e_{c,j_2}$ となるような $\theta_{c,j_1}, \theta_{c,j_2}$ の設定が与えられた場合を考える. θ_{c,j_2} を微小量 Δ 増加し, θ_{c,j_1} を微小量 Δ 減少させることで, L1 ノルム上限の変更前と比較して $(e_{c,j_2} - e_{c,j_1})\Delta$ だけ予測誤差の二乗

-
- 1: Initialize $\mathbf{a}_c^{(j)} \leftarrow \mathbf{0}$ for all c, j
 - 2: Initialize $\theta_{c,j} \leftarrow 0$ for all c, j
 - 3: Initialize $e_{c,j}$ for all c, j
 - 4: **while** $\frac{\sum_c \|A_{c,j}\|_0}{NMC} \leq r$ **do**
 - 5: $c', j' \leftarrow \arg \max_{c,j} e_{c,j}$
 - 6: Increase $\theta_{c',j'}$
 - 7: $\mathbf{a}_{c'}^{(j')}$ \leftarrow solution to eq. (3)
 - 8: Update $e_{c',j'}$ with eq. (4)
 - 9: **end while**
-

図 5: 予測係数行列設計問題の求解法

和を削減できるため、前述の L1 ノルム上限は最適でない。対称性より $e_{c,j_1} > e_{c,j_2}$ の場合も同様に最適でない。従って、 $e_{c,j_1} = e_{c,j_2}$ に対応する L1 ノルム上限が最適である。

予測誤差削減効率の均等化は、以下の方針に基づき行う。予測誤差削減効率が最大となる単画素予測係数ベクトル設計問題を選択し、L1 ノルムを増加させる。増加量の具体的な設定方法は、図 5 の説明において後述する。予測誤差削減効率が L1 ノルム上限に関して単調減少するため、L1 ノルム上限の増加により予測誤差削減効率が減少する。この処理を全予測係数の非零値が指定された割合となるまで繰り返す。本反復処理により、予測誤差削減効率が近似的に均等化される。

提案手法では予測誤差削減効率を近似的に均等化する方針の下、図 5 の求解法により、式 (2) の予測係数行列設計問題を解き、その近似解を出力する。1 行目では全ての単画素予測係数ベクトル $\mathbf{a}_c^{(j)}$ を零値に初期化する。2 行目では、L1 ノルム上限を零値に設定する。3 行目では式 (4) で定義される予測誤差削減効率を計算する。単画素予測係数ベクトルの更新は予測係数全体の非零値の係数の割合が r になるまで、5 行目から 8 行目を続ける。ここで、 $\|A_c\|_0$ は予測係数行列内における非零値の予測係数の個数を表す。予測係数全ての要素数は NMC 個であるため、4 行目における条件式の左辺は予測係数全体の非零値の係数の割合を表す。5 行目では、予測誤差削減効率が最も大きなクラス番号 c' 、被予測画素 j' を選択する。6 行目では、選択された c', j' に対応する L1 ノルム上限 $\theta_{c',j'}$ を増加させる。 $\theta_{c',j'}$ の増加は (1) $\mathbf{a}_{c'}^{(j')}$ 内のいずれかの零値の予測係数が非零値となる直前、(2) $\mathbf{a}_{c'}^{(j')}$ 内のいずれかの非零値の予測係数が零値となる、のいずれかを満たすまで行う。7 行目では、L1 ノルム上限を θ_n に設定し、式 (3) により単画素予測係数ベクトルを定める。8 行目では、7 行目で求めた単画素予測係数ベクトルに対して予測誤差削減効率を再計算する。

2.5 クラス番号および予測係数行列の表現方法

クラス番号の表現には、修正した Most Probable Mode (MPM) を利用する。符号化対象予測ブロックでは、近傍の予測ブロックにおけるクラス番号および、予測係数間の差分における係数の二乗和に基づき、MPM の候補リストが構築される。当該予測ブロックの上、左上、左の近傍予測ブロックで選択されたクラス番号をそれぞれリストの 1 番目、2 番目、3 番目へ登録する。重複したクラス番号は登録しない。候補リスト内のクラス番号が 3 つに満たない場合は、最頻出のクラスが有する予測係数行列と二乗誤差の観点で近い順、かつ重複しないように、リスト内にクラス番号を登録する。予測係数は (1) 予測係数が零値か否かを表わすフラグ、(2) 32bit の浮動小数点数で表される予測係数、の 2 種類の情報により表現する。

3 実験

提案手法の有効性を検証するため、提案手法を HEVC 参照ソフトウェアのオリジナルの HM16.7 へ実装し、ロスレス圧縮に要する符号量を HM16.7 と比較した。符号化対象として、5 種類の自然画像 (解像度: 3840 × 2160 [画素]、色差フォーマット: YCbCr4:2:0、ビット深度: 8 [bit/channel]) の先頭 50 フレームを用いた。提案手法における予測ブロックは PU として表現し、PU サイズは 8 × 8 [画素] とした。クラス番号は PU 単位で設定し、CU サイズは 8 × 8 [画素] 固定、PU 分割なし、TU サイズも 8 × 8 [画素] 固定としている。その他の実験条件は、文献 [14] に従った。クラス数 C は候補値 (11, 15, 19, ..., 39) の中から各符号化対象の先頭フレームにおける発生符号量を最小化する値を設定した。被予測画素数 N は 64、参照画素数 M は 25 とした。非零値の予測係数の割合 r は候補値 (10%, 15%, 20%, ..., 40%) の中から各符号化対象の先頭フレームにおける発生符号量を最小化する値を設定した。クラス番号と予測係数の交互最適化は 20 回行った。

結果を表 1 に示す。提案手法は HM16.7 と比較して平均 1.39%、最大 2.24% の符号量削減が確認できる。この理由は提案手法により予測残差が低減し、残差表現に要する符号量を削減できたためであると考えられる。また、HEVC と比較して最も削減符号量が多かった Ducks take off は、曲線状に波紋が広がる映像であり、HEVC の方向性予測では正確な予測が困難であったため、提案手法での削減符号量が大きかったと考えられる。

次に、提案手法におけるクラス数、非零値の係数の割合に関する考察を行う。表 2 は、各符号化対象映像の先頭フレームを圧縮した際に最も符号量が小さいクラス数、非零値の係数の割合を示している。表 2 から、高圧縮な符号化が困難な映像ほど使用される予測係数の数が多いことがわ

表 1: HM16.7 と提案手法における符号量の比較および削減率 .

符号化対象	HM16.7 [kB]	提案手法 [kB]	削減率 [%]
Crowd Run	7173	7072	1.41
Ducks take off	7756	7582	2.24
In to tree	6796	6718	1.14
Old Town Cross	7355	7281	1.01
Park joy	6452	6380	1.12
平均	7106	7007	1.39

表 2: 提案手法における最小符号量を達成するクラス数および非零値の予測係数の割合 .

符号化対象	符号量 [kB]	クラス数 [個]	非零値の割合 [%]
Crowd Run	7072	35	20
Ducks take off	7582	31	30
In to tree	6718	19	35
Park joy	7281	31	30
Old town cross	6380	35	20

かる . その理由は , 予測係数行列に含まれる非零値の係数の個数は , クラス数と非零値の係数の割合の積に比例するためである .

4 おわりに

本稿では , 予測器を映像の局所的な信号特性へ適合させるため , 画像ごとに複数の線形予測器を適応的に構築する新たなイントラ予測方式を提案した . 提案手法では , 予測器を表現する上で付加情報となる予測係数情報の効率的な圧縮を実現するため , 予測係数にスパース性の制約を課し , 符号化効率の良い係数を導出するための新たな最適化手法を提案した . 実験の結果 , ロスレス圧縮において HEVC 参照ソフトウェア HM16.7 と比較して , 平均 1.39% , 最大 2.24% の符号量削減を確認した . 今後の課題は , 予測係数の符号化効率の改善 , ロッシー符号化への適用等が挙げられる .

参考文献

- [1] High efficiency video coding. *ITU-T Rec. H.265 and ISO/IEC 23008-2*, May 2015.
- [2] S. Matsuo, S. Takamura, and A. Shimizu. Intra angular prediction with weight function and modification filter. *Picture Coding Symposium*, pp. 77–80, Dec. 2013.

- [3] E. Alshina, A. Alshin, J. Min, K. Choi, A. Saxena, and M. Budagavi. Known tools performance investigation for next generation video coding. *ITU-T SG16/Q6 document VCEG-AZ05*, June 2015.
- [4] 中馬高明, 松田一朗, 亀田裕介, 伊東晋. 円弧に沿った両方向イントラ予測の検討. 画像符号化シンポジウム (PCSJ) P-2-05, Nov. 2015.
- [5] C. Chen, X. Xu, R. Liao, W. Peng, S. Liu, and S. Lei. Screen content coding using non-square intra block copy for HEVC. *IEEE International Conference on Multimedia and Expo*, pp. 1–6, July 2014.
- [6] T. Chang, C. Chen, R. Liao, C. Kuo, and W. Peng. Intra line copy for hevc screen content coding. *Signal and Information Processing Association Annual Summit and Conference*, pp. 1–8, Dec. 2014.
- [7] X. Xu, S. Liu, J. Ye, and S. Lei. PU level intra block copying with flipping mode. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 1–7, Dec. 2014.
- [8] Z. Zhang and V. Sze. Rotate intra block copy for still image coding. *IEEE International Conference on Image Processing*, pp. 4102–4106, Sep. 2015.
- [9] Y. He, X. You, Q. Wang, and D. Xu. Adaptive intra prediction filtering(AIPF). *IEEE International Conference on Security, Pattern Analysis, and Cybernetics*, pp. 358–362, Oct. 2014.
- [10] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, Vol. B-58, No. 1, pp. 267–288, 1996.
- [11] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, Vol. 32, No. 2, pp. 407–499, 2004.
- [12] W. Fu. Penalized regressions: The bridge versus the LASSO. *Journal of Computational and Graphical Statistics*, Vol. 7, No. 3, pp. 397–416, 1998.
- [13] B. Amir and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, Vol. 2, No. 1, pp. 183–202, Mar. 2009.
- [14] F. Bossen. Common test conditions and software reference configurations. *ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC) document JCTVC-L1100*, ITU-T and ISO/IEC JCT, Geneva, Jan. 2013.