

DCT 係数のサインビットの符号化において 高い圧縮効率と計算効率を同時に達成するには？

How Can We Achieve High Compression and Computational Efficiency for Coding Sign Bits of DCT Coefficients?

伊藤 楓馬* 都竹 千尋* 高橋 桂太* 藤井 俊彰*
Fuma Ito Chihiro Tsutake Keita Takahashi Toshiaki Fujii

1 はじめに

離散コサイン変換 (discrete cosine transform, DCT) に基づく画像符号化において, DCT 係数のサインビット (以降, 単にサイン) はブロック内で相関が確認できず, その圧縮は不可能であると考えられていた. 圧縮の手法として, サインの歪を許容した sign data hiding [1] が提案されているが, 本研究ではサインのロスレス符号化を目的とする. 文献 [2] では, 同一周波数に属する DCT 係数のサインは, ブロック間で相関があることが示されている. この事実に基づいて, 文献 [3, 4] では, サインの復元を介して圧縮するアプローチを採用しており, サイン付きの係数をエントロピー符号化する代わりに, サインの復元誤差と振幅を符号化する. 復元精度が高いほど誤差は 0 に偏るため, 効率的な圧縮が可能となる.

文献 [3, 4] の枠組みに基づいて, 著者らはこれまでに, DCT 係数の振幅のみから高精度にサインを復元する sign retrieval (SR) を提案している [5, 6, 7]. これは, 位相回復—DFT 係数の振幅のみから位相を復元する問題—から着想を得た手法である. 文献 [5] では, 凸解析に基づく convex sign retrieval (CSR) を提案している. CSR は画像復元を介した手法であり, 非線型写像の反復によって復元画像の精度を高める必要がある. 従って, 計算量が膨大である. 文献 [6] では, 画像復元を深層展開した unrolled sign retrieval (USR) を提案している. USR では, CSR の画像復元処理を深層ニューラルネットワークに置き換えることで, 高い圧縮効率を達成した. しかし, CSR と同様に計算量の課題が残されていた.

この課題を解決するために, 著者らは振幅からサイン

を直接的に復元する direct sign retrieval (DSR) を提案している [7]. CSR と USR では, 画像復元を介してサインを復元しており, その過程で DCT を実行する必要がある. 一方, DSR は画像復元が不要であり, DCT も一切必要としない手法である. CSR や USR と比較して, DSR は圧倒的に少ない計算量でサインを復元することが可能である. しかし, 圧縮効率の面で課題が残されていた. 本稿では, 圧縮効率と計算効率の両立を念頭に, DSR のアルゴリズムを最適化する. この最適化により, USR と CSR を上回る圧縮効率を, 極めて少ない計算量で達成できたことを報告する.

2 Sign Retrieval に基づくサイン圧縮

2.1 エンコーダ/デコーダ

本節では, 文献 [5, 6, 7] のエンコーダとデコーダを概観し, 符号化対象画像を復号するパイプラインを述べる. エンコーダとデコーダを導入するベースライン手法として, JPEG を採用する.

エンコーダ

1. DCT 係数を計算し, 量子化を適用.
2. 振幅とサイン^{*1}に分割し, 振幅を符号化+伝送.
3. 後述する SR を介して, サインをローカル復元.
4. 原サインと復元サインの XOR を計算し, 誤り訂正ビット列を得る.
5. 誤り訂正ビット列を符号化+伝送.

デコーダ

1. DCT 係数の振幅を復号.
2. SR (エンコーダと共有) を介して, サインを復元.

* 名古屋大学 大学院工学研究科 情報・通信工学専攻

*1 以降, 量子化後の係数に付随するサインを原サインと呼ぶ.

3. 復元サインと誤り訂正ビット列の XOR を計算し、原サインを復元.
4. 振幅と原サインから、画像を復号.

SR によるサインの復元には誤差が発生する可能性があるが、誤り訂正ビット列を送信することで、原サインをロスレス符号化/復号することが可能である. このとき、サインの復元精度が高いほど誤差は 0 に偏るため、効率的な圧縮が可能となる.

2.2 Convex/Unrolled SR

位相回復は、DFT 係数の振幅からその位相を復元する問題である. DFT 係数の位相 α を 0 または π に限定した場合、 $e^{j\alpha} = \pm 1$ となる. 従って、サインは位相の特殊ケースであると解釈することができる. 位相回復の方法論に基づいて、SR では DCT 係数の振幅からサインを復元する.

高さ h と幅 w がそれぞれ h, w の画像を JPEG で符号化し、得られた画像を $X \in \mathbb{R}^{h \times w}$ とする. ブロックサイズは 8×8 である. JPEG 画像 X の m, n 番目のブロックを $X_{m,n} \in \mathbb{R}^{8 \times 8}$, その DCT 係数を $Y_{m,n} \in \mathbb{R}^{8 \times 8}$ とすると、SR は次のよう定式化することができる.

$$\text{find } S_{m,n} \text{ s.t. } S_{m,n} \odot |Y_{m,n}| = Y_{m,n} \quad (1)$$

ここで、 $S_{m,n} \in \{-1, +1\}^{8 \times 8}$ は $Y_{m,n}$ のサインであり、 \odot は要素ごとの積である. 一般に、 $Y_{m,n}$ は未知であり、解は一意的に定まらない.

一方、ブロック $X_{m,n}$ そのものを復元し、復元ブロック $\hat{X}_{m,n} \in \mathbb{R}^{8 \times 8}$ の DCT 係数からサインを得ることも考えられる. 解を一意化するために、式 (1) に対して正則化関数 f を導入すると、SR は以下の最小化問題に帰着される.

$$\min_{\hat{X}_{m,n} \in \mathbb{R}^{8 \times 8}} f(\hat{X}_{m,n}) \text{ s.t. } |\text{DCT}(\hat{X}_{m,n})| = |Y_{m,n}| \quad (2)$$

この問題を全ての m, n に対して解く必要がある. 単一ブロックのサインは最大 2^{64} 通りの組み合わせが存在するため、この計算は現実的ではない.

この計算量の問題を解決するために、CSR では、文献 [8] の位相回復アルゴリズムを応用し、方程式を以下のように不等式化している. また、JPEG 画像 X を復元することで、全てのブロックを並列に復元する.

$$\min_{\hat{X} \in \mathbb{R}^{h \times w}} f(\hat{X}) \text{ s.t. } |\text{DCT}(\hat{X}_{m,n})| \leq |Y_{m,n}|, \forall m,n \quad (3)$$

この不等式化により、組合せ最適化問題が連続最適化問題に置き換わる. また、適切な正則化関数 f を採用する



(a) 振幅 (黒:小, 白:大) (b) サイン (黒:負, 白:正)

Fig.1: DCT 係数の 0/1 周波数成分を集めたサブバンド

ことで、凸解析の手法を用いて解くことが可能であり、現実的な計算量で式 (2) の近似解を求められる.

CSR では、正則化関数にウェーブレット変換の l_1 ノルムを採用している. 式 (3) の求解には、ウェーブレット縮退処理と DCT 係数の凸射影の反復を行う. CSR の反復回数は 200 回が推奨されている. USR では、正則化に畳み込みニューラルネットワーク (CNN) を使用している. 従って、CNN による推論と凸射影の反復を行う. USR の反復回数は 20 回が推奨されている. CSR と USR は非線形写像を繰り返す反復解法であるため、膨大な計算量が必要である.

2.3 Direct SR

図 1 は、水平/垂直周波数インデックスが 0/1 の DCT 係数の振幅とサインである. このように、同一の周波数に属する DCT 係数を集めた二次元信号を、本稿ではサブバンドと呼ぶ. 図 1(a) は振幅分布であり、空間的な相関を持つことが確認できる. また、図 1(a) と (b) を比較すると、振幅が不連続となる領域とサインが不連続となる領域が概ね一致している. これらの対応に基づいて、著者らはサブバンド内の振幅の大小関係からサインを復元する DSR を提案している.

DSR では、サブバンド内の相関を捉えるために CNN を使用する. また、サブバンドの振幅分布からサインを直接的に復元する. ブロックサイズとして 8×8 を採用した場合、一つのサブバンドのサイズは $h/8 \times w/8$ となり、計 64 種類が得られる. これらをチャンネル方向にスタックしたものを $Y_X \in \mathbb{R}^{64 \times h/8 \times w/8}$ と記述する. Y_X の振幅とサインを各々振幅テンソル $|Y_X|$ 及びサインテンソル S_X と呼ぶ. DSR では、次式のロジスティック回帰によって $64 \times h/8 \times w/8$ の確率テンソルを推論する.

$$P_X = \phi_\theta(|Y_X|) \quad (4)$$

ここで、 ϕ_θ はパラメータ θ を持つ CNN である. $P_X \in [0, 1]^{64 \times h/8 \times w/8}$ は、 $|Y_X|$ の原サインが正である確率を

表す。サインの復元は最尤推定によって行われ、 P_X が 0.5 以上である場合は正、0.5 未満では負を採用する。この CNN の学習には、振幅テンソルとサインテンソルが用いられ、その詳細は 4 章で示す。

DSR は、CSR や USR のような画像復元処理や反復を行わずに、サインを直接的に復元することができる。DSR では、少ない計算量でサインを得られるが、復元精度は CSR/USR が上回る。

3 圧縮効率と計算効率を両立するには？

誤り訂正ビット列を少ない符号量で伝送するには（高い圧縮効率を得るには）、サインを高精度に復元する必要がある。本稿では、DSR の枠組みを踏襲しつつ、圧縮効率と計算効率の両立を目指す。そこで、学習用画像、ネットワークのレイヤー数、及びパラメータ数を最適化する。但し、これらを同時に最適化することは不可能であるため、独立に最適化を行う。

学習用画像

JPEG では、圧縮時の品質を表すパラメータである quality factor (qf) を指定し、qf に依存して係数の量子化が行われる。このとき、qf が小さいほど量子化された係数は 0 を多く含むようになる。この事実に対して、DSR では学習用の振幅テンソルとして量子化前のものを用いている。これにより、CNN は推論を行う際に「入力の DCT 係数は 0 の個数が少ない」ことを暗黙的に仮定する。qf が小さい場合ではこの仮定に反するため、圧縮効率の低下を招く恐れがある。学習時と推論時の qf を変化させて、復元結果にどれほどの影響を与えるかを確認する (4.1 節)。

レイヤー数

表 1 に DSR のネットワーク構造を示す。ストライドは全て 1 である。Conv-1 の入力と Conv-2 の出力は、各々振幅テンソル $|Y_X|$ と確率テンソル P_X である。DSR は計算効率を重視しているため、ネットワークのレイヤー数は最小限であった。高い圧縮効率を実現するために、ネットワークのレイヤー数と圧縮効率のトレードオフを確認し、レイヤー数を最適化する (4.2 節)。

パラメータ数

圧縮効率を向上させるためにネットワークのレイヤー数を増加させると、レイヤー数と共に計算量も増加する。そこで、圧縮効率を維持しつつ、ネットワークのパラメータ数を削減する方法について考える。DSR は低周

Table.1: DSR [7] のネットワーク

Layer	Kernel size	In/Out ch.	Act.
Conv-1	3×3	64/64	ReLU
Conv-2	3×3	64/64	Sigmoid

Table.2: 4 章のネットワーク (隠れ層の数は変動)

Layer	Kernel size	In/Out ch.	Act.
Input	3×3	64/128	ReLU
Hidden	3×3	128/128	ReLU
Output	3×3	128/64	Sigmoid

波成分のサインの復元精度が高い一方、高周波成分のサインを高精度に復元することが困難である。しかし、高周波成分の係数は量子化によって零に丸められ、非零係数のサインの圧縮を省略しても、ビットストリーム全体の圧縮効率に及ぼす影響は僅かである。そこで、高周波成分のサインの学習と復元を省略することで、パラメータ数を削減する。また、低周波成分のサインの復元精度を向上させる副次的効果が期待され、高周波成分の復元を省略したとしても、周波数全体にわたる圧縮効率の変動は微小であると予想される。高周波成分に対応する誤り訂正ビットの符号化では、単位ビットあたり 1 ビットを割り当てることを仮定する (4.3 節)。

4 最適化実験

評価指標

圧縮効率を定量的に確認するために、誤り訂正ビット列のエントロピーを評価する。サブバンド毎にエントロピーを計算し、チャンネルにわたるエントロピーの期待値を計算した。これは、チャンネル毎に独立の確率モデルを割り当てたときに、単一のサインを符号化する際に必要なビット数 (bits per sign, bps) の理論下限を意味する。なお、サインビットが自明である直流成分、及び DCT 係数の絶対値が 0 のサインビットは、エントロピーの計算から除外した。

学習方法

学習用画像に対して DCT と量子化を適用し、振幅テンソル $|Y_X|$ とサインテンソル S_X を計算する。サインテンソル S_X と CNN から得られる確率テンソル $P_X = \phi_\theta(|Y_X|)$ の binary cross entropy H を最小化する

Table.3: 学習時とテスト時の qf とエントロピー

test qf	train qf			
	20	50	80	100
20	0.753	0.746	0.756	0.766
50	0.829	0.822	0.823	0.836
80	0.893	0.893	0.879	0.892
100	0.984	0.986	0.985	0.979

るように、パラメータ θ を最適化する.

$$\min_{\theta} \sum_{X \in D} H(\phi_{\theta}(|Y_X|), S_X) \quad (5)$$

ここで、 D は学習用画像のデータセットである. 本稿では、CLIC 2020^{*2}で配布されている学習用画像を $h \times w = 256 \times 256$ にクロップしたもの、計 6,532 枚を D とした. 一方、テスト画像として CLIC 2021 で配布されている 60 枚を使用した. 表 2 にネットワーク構造を示す. 隠れ層の入力/出力チャンネルは全て 128/128, カーネルサイズは 3 で固定した. 以降の実験は、隠れ層の数を変化させながら行った.

4.1 学習用画像の最適化

学習用画像とテスト画像の DCT 係数を様々な qf で計算し、学習時とテスト時の qf が異なる場合のサインの復元精度を確認する. ここでは、隠れ層の数を 2 とした. 学習用画像の qf として 20, 50, 80, 100 を採用し、各 qf で最適化した 4 種の CNN を用意した. 同様に、テスト時の qf として 20, 50, 80, 100 を採用し、各 qf で量子化した振幅テンソルを学習済みの各 CNN に入力した. 従って、単一のテスト画像毎に、計 $4 \times 4 = 16$ のエントロピーを算出した. 表 3 に結果を示す. この数値は、テスト画像にわたるエントロピーの平均値である.

まず、テスト時の qf を固定して学習時の qf を変化した場合、qf 20 の場合を除き、学習時とテスト時の qf が近いほど低いエントロピーが得られる. 一方、qf が異なるとしても、エントロピーが大幅に増加するという事は確認されない. 従って、学習時の qf を基準として、テスト時の qf に差異があったとしても、サインの復元精度にほとんど影響を与えないと結論づけられる.

次に、学習時の qf を固定してテスト時の qf を変化した場合、テスト時の qf が小さくなるにつれて、エン

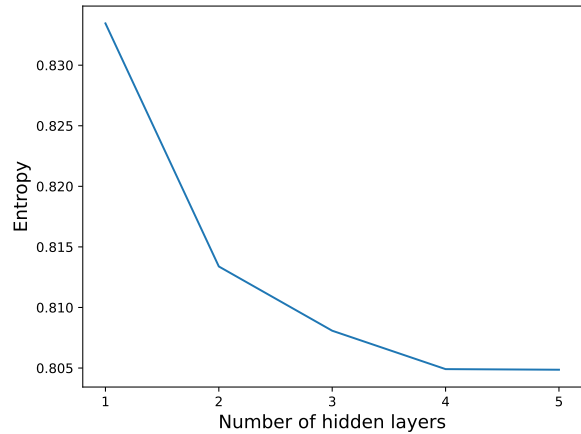


Fig.2: レイヤー数とエントロピー

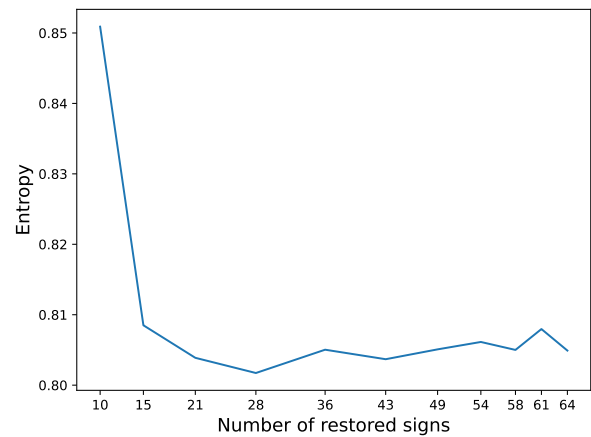


Fig.3: 復元するサインの数とエントロピー

トロピーが減少している. 一般に、qf が小さいほど、高周波成分の非零係数の数が少なくなり、低周波成分のサインの数が相対的に多くなる. 従って、エントロピーは低周波成分のサインの復元精度に大きく依存する. DSR は、低周波成分のサインを高精度に復元できる性質を持つため、エントロピーが減少する結果が得られたと推察される.

最後に、学習済みの CNN の汎化性能を確認する. テスト時の qf として 5, 10, ..., 90, 95 の計 19 種類を採用した. qf 20, 50, 80, 100 で学習した CNN を介して得られたエントロピーは、平均で 0.821, 0.817, **0.815**, 0.826 であり、qf 80 で学習した CNN が最も高い汎化性能を持つことが確認された. そこで、以降の実験では全ての CNN を qf 80 で最適化する.

4.2 レイヤー数の最適化

ネットワークのレイヤー数を変化させて圧縮効率を確認する. 隠れ層の数を 1 から 5 まで変化させ、5 種類

^{*2} <http://compression.cc>

の CNN を最適化した。テスト時には、5-95 の計 19 種類の qf を指定してサインを復元した。図 2 に各 CNN を介して得られたエントロピーを示す。横軸は隠れ層の数、縦軸はエントロピーを表す。ここでは、19 通りの qf にわたるエントロピーの平均値をプロットしている。隠れ層の数が 4 層までは、層を増やす毎にエントロピーが減少していることが確認できるが、4 層と 5 層では大きな差が見られない。従って、隠れ層は 4 層あれば十分な復元精度が得られると結論づけられる。

4.3 パラメータ数の最適化

高周波成分のサインの復元を省略して、パラメータ数を削減した場合の圧縮効率を確認する。ここでは、隠れ層の数を 4 とした。DCT 係数をジグザグスキャン順で番号付けし、サインの復元を先頭から 10, 15, 21, 28, 36, 43, 49, 54, 58, 61, 64 個に制限する。即ち、54, 49, 43, 36, 28, 21, 15, 10, 6, 3, 0 個のサイン復元を省略した。入力層と出力層のチャンネル数はこれらの数値に合わせた。テスト時の qf は 5-95 の 19 通りとした。エントロピーの期待値を公平に評価するために、復元を省略したサインに対応する誤り訂正ビット列のエントロピーは 1 とした。図 3 に復元するサインの個数とエントロピーの関係を示す。サインの復元を先頭から 28 個に制限した場合、即ちサインの復元を 36 個省略した場合に最も低いエントロピーが得られている。この事実から、高周波成分のサインの学習を省くことで、低周波成分のサインの復元精度を向上させることができたと推察される。以上より、パラメータ数を削減した上で、圧縮効率を維持することが達成された。

5 比較実験

以上の最適化実験に基づいて、表 4 に圧縮効率と計算効率を両立する CNN を示す。以降、この最適化後の DSR を CSR [5], USR [6], 及び最適化前の DSR [7] と比較する。最適化前の DSR, 最適化後の DSR, USR の CNN は各々 qf 100, 80, 50 で学習した。

5.1 圧縮効率の比較

テスト時の qf として 5-95 の計 19 種類を採用し、圧縮効率を比較する。図 4 に各手法の誤り訂正ビット列のエントロピーを示す。CSR では 0.665-0.952, USR では 0.610-0.941, 最適化前の DSR では 0.675-0.954 である。最適化後の DSR では 0.615-0.949 であり、 qf 5 や 95 などの極端な場合を除いて、最も低いエントロピーを

Table.4: 最適化後の DSR

Layer	Kernel size	In/Out ch.	Act.
Conv-1	3×3	28/128	ReLU
Conv-2	3×3	128/128	ReLU
Conv-3	3×3	128/128	ReLU
Conv-4	3×3	128/128	ReLU
Conv-5	3×3	128/28	Sigmoid

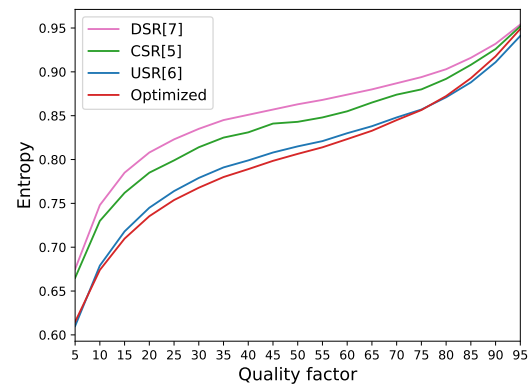


Fig.4: qf と各手法のエントロピー

達成している。

5.2 計算効率の比較

各手法の圧縮効率と計算効率を比較する。表 5 に本実験で用いた計算機環境を示す。CNN の推論を含む全ての計算は、CPU で行った。テスト時の qf として 5-95 の計 19 種類を採用し、 qf にわたるエントロピーの平均値で評価する。図 5 は各手法で得られた誤り訂正ビット列のエントロピーと計算時間である。横軸は画像 1 枚あたりの全サインビットを復元するために必要な時間、縦軸は誤り訂正ビット列のエントロピーを表す。反復手法である CSR 及び USR については、それぞれ反復 1 回目と 200/20 回目の 2 点をプロットしている。USR は高い圧縮効率を達成できているが、計算量が膨大である。また、最適化前の DSR は、計算量が大幅に削減されているものの、圧縮効率は不十分である。一方、最適化後の DSR は、少ない計算量と高い圧縮効率を同時に達成できている。最適化後の DSR は、CSR 及び USR の反復 1 回目に比べても計算量が少なく、かつ反復 20 回目の USR の圧縮効率に匹敵している。

5.3 復元画像の比較

CSR, USR, 最適化前の DSR, 及び最適化後の DSR で得られた復元サインを DCT 係数の振幅に乗算し、逆

Table.5: 計算機環境

CPU	Intel Core i7-6700K
Main memory	32 GB
OS	Ubuntu 20.04 LTS
Language	Python 3.9.12
Framework	PyTorch 1.13.1

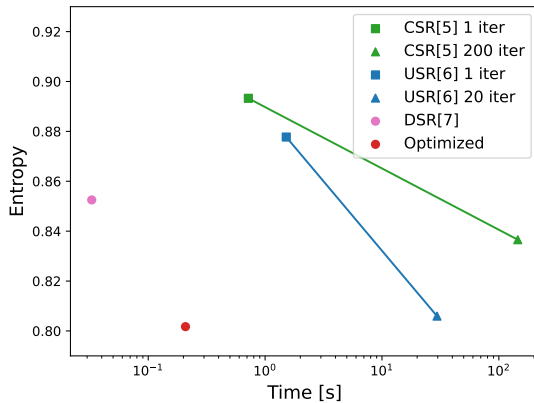


Fig.5: 計算時間とエントロピー

DCT を適用して得られる復元画像の品質を評価した。図 6 は各手法の復元画像である。(a) は qf を 80 とした JPEG 画像である。(b)–(e) は各手法で得られた復元画像であり、併記した数値は (a) と復元画像間の PSNR を表している。(b) CSR や (c) USR では、画像復元処理を介するため、ブロック雑音のない画像を復元できている。対して反復や画像復元処理を含まない (d) DSR では、ブロック雑音が顕著に見られる。(e) 最適化後の DSR では、(d) と比較してブロック雑音が低減されており、より自然な画像を再構成できていることがわかる。復元サインの精度が高ければ、高画質な復元画像が得られると考えられるが、前章で述べたように、この直感に反して最適化後の DSR が最も低いエントロピーを与えている。これは、画質に大きく寄与しない周波数成分のサインを高精度に復元できているためと推察される。

6 まとめ

本稿では、DCT 係数のサインビットを CNN によって直接復元する DSR に対し、圧縮効率や計算効率に影響する要素や特性を精査した。最適化後の DSR では、最適化前の課題であった圧縮効率と計算効率の両立が達成された。今後の展望としては、誤り訂正ビット列の具体的な符号化方法の検討が挙げられる。



Fig.6: 復元画像の比較

参考文献

- [1] G. Clare, F. Henry, and J. Jung, “Sign data hiding,” *JCTVC-G271*, 2011.
- [2] C. Tu and T. D. Tran, “Context-based entropy coding of block transform coefficients for image compression,” *IEEE Trans. Image Process.*, vol. 11, no. 11, pp. 1271–1283, 2002.
- [3] O. Miroshnichenko, M. Ponomarenko, V. Lukin, and K. Egiazarians, “Compression of signs of DCT coefficients for additional lossless compression of JPEG images,” in *Proc. Int. Symp. Electronic Imaging: Image Processing: Algorithms and Systems XVI*, 2018.
- [4] A. Nakagawa, J. Koyama, K. Kazui, and J. Katto, “Sign information predictive coding of transform coefficients in video compression,” *IEICE Trans. Inf. & Syst.*, vol. J100-D, no. 9, pp. 819–830, 2017.
- [5] C. Tsutake, K. Takahashi, and T. Fujii, “An efficient compression method for sign information of DCT coefficients via sign retrieval,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2021.
- [6] K. Suzuki, C. Tsutake, K. Takahashi, and T. Fujii, “Compressing sign information in DCT-based image coding via deep sign retrieval,” *arXiv:2209.10712*, 2022.
- [7] F. Ito, C. Tsutake, K. Takahashi, and T. Fujii, “A method for fast compression of sign bits of DCT coefficients in image coding,” *IEICE Tech. Rep.*, vol. 122, no. 439, pp. 178–181, 2023.
- [8] T. Goldstein and C. Studer, “PhaseMax: convex phase retrieval via basis pursuit,” *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2675–2689, 2018.