

学習型静止画像圧縮の実用化に関する研究

研究代表者

孫 鶴鳴

横浜国立大学 大学院工学研究院 准教授

1 はじめに

IoT 社会において、画像の伝送や保存の負担を軽減するために、画像圧縮技術は非常に重要である。従来の画像圧縮規格は、30 年以上にわたって開発されてきた。一方、ニューラルネットワーク技術の発展に伴い、学習型画像圧縮 (LIC) [1][2][3][4][5][6] は、最先端の動画像圧縮規格のイントラ予測[7] を上回る符号化性能を示している。しかし、LIC は高い圧縮性能を有するものの、以下の課題が残されており、現時点では実用化に向けてさらなる検討が必要である。1) 演算に膨大な時間を要し、リアルタイム処理に対応できない。2) 可変ビットレートを実現するには、複数のニューラルネットワークモデルが必要になる。3) 敵対的攻撃に対して脆弱であり、セキュリティ面の検討が不十分である。

本研究プロジェクトでは、第 2 章において LIC のスループット向上について述べる。続いて、第 3 章では復号計算量の削減を目的とした過学習型 LIC を提案する。さらに、第 4 章では可変ビットレート LIC を提案し、第 5 章では LIC に対する敵対的攻撃および防御の効果について検討する。

2 LIC のスループット向上

2.1 研究背景

LIC の高速化にはハードウェア開発が必要であり、これまでもいくつかの先行研究 [8][9] が報告されている。しかし、これらの研究は既存のアルゴリズムを前提としてハードウェアを開発しているため、スループット性能の向上には限界がある。本研究では、アルゴリズムとアーキテクチャの協調最適化により、スループットの向上を目指す。

2.2 ニューラルネットワーク構造

本研究で用いる学習型画像圧縮の基本フレームワークについて説明する。本研究で用いる基本フレームワークでは、主経路は、入力画像 x を潜在表現 y に変換する分析変換 g_a と、量子化された潜在表現 $\hat{y}$ から再構成画像 $\hat{x}$ を生成する合成変換 g_s から構成される。ここで、 g_a および g_s はそれぞれニューラルネットワークで構成されている。

また、潜在表現 y の確率分布をより正確に推定するために、ハイパーパスも導入されている。ハイパーパスでは、 $\hat{y}$ からハイパー潜在表現 z を生成し、さらに量子化された $\hat{z}$ から y の事前分布のパラメータである平均 μ_y および標準偏差 σ_y を推定する。本研究では、 y が単一ガウス分布に従うと仮定し、さらに一様分布との畳み込みにより量子化の影響を考慮する。一方、 z の事前分布については、従来研究と同様に非パラメトリックな因子化モデルを用いて表現する。

最終的な学習目的関数は、レート歪み最適化に基づいて定義され、符号量 R と歪み D の重み付き和 $J = R + \lambda D$ として表される。ここで、 R は y および z の符号量から構成され、 D は原画像 x と再構成画像 $\hat{x}$ の歪みを表す。

2.3 FPGA アーキテクチャ

入力バッファは DDR メモリから入力画像を読み込むために使用され、出力バッファは最終的な処理結果を DDR メモリへ書き戻すために使用される。各層は、活性化バッファ、重みバッファ、および処理要素 (PE) アレイで構成される。活性化バッファは前層の計算結果を一時的に保持し、重みバッファは DDR メモリから必要な重みを読み込む。PE アレイは主に乗算および加算処理を担当する。本アーキテクチャでは、カーネル幅方向、入力チャネル方向 c 、出力チャネル方向 m の 3 つの次元における並列性を利用する。入力チャネル C 、出力チャネル M 、およびカーネル高さ方向をすべて走査することで、1 つの出力要素が生成される。

2.4 チャンネル並列度の制約

配線混雑により、実際の DSP 使用数は、FPGA ボードに搭載されている DSP 数よりも少なくなる。従来研究では、KU115 を用いて 720P 画像の符号化を行う場合、5520 個の DSP のうち 2654 個しか利用できなかった。この問題を緩和するため、本研究ではチャンネル方向の並列度に制約を導入する。

具体的には、2 種類の制約を検討する。1 つは、 c および m を 2 のべき乗に制限するハード制約であり、もう 1 つは、 c および m を 2、4、または 8 の倍数に制限するソフト制約である。実験結果に示すように、最終的な制約条件はスループットに基づいて選択される。

チャンネルの並列性に制約を加えることで、DSP の利用率を向上させる。配置配線フェーズでボトルネックの一つである活性化バッファの LUT を削減することで、配線エラーを発生させることなく、より多くの DSP を使用することができる。

2.5 ニューラルネットワーク探索

チャンネル並列度に制約を加えることで、ハードウェアコストを削減することができる。一方で、ニューラルネットワークモデルが固定されている場合、チャンネル並列度のような粗い粒度で DSP を割り当てると、DSP 効率が低下する可能性がある。

DSP 効率を向上させるため、本研究では全体のチャンネル数 C および M を調整する。デフォルトのチャンネル数を基準として、最初の 4 つの ga 層の出力チャンネル数を探索する。チャンネル並列度 c および m を選択し、その後 DSP 効率を算出する。最終的に、DSP 効率が最も高いネットワークを選択する。複数の探索結果が同一の DSP 効率を示す場合には、BRAM 使用量が最も小さいものを選択する。なお、デコーダ部への影響を避けるため、最後の ga 層の出力チャンネル数は変更しない。

ニューラルネットワーク探索により DSP 効率を向上させる。各レイヤーの全体的なチャンネルを調整することで、割り当てられた DSP はより高い効率（すなわちクロックサイクルのバブルが少ない）で動作することができる。

2.6 実装の詳細

アルゴリズムについて、まず、TensorFlow を用いて浮動小数点モデルをゼロから学習する。損失関数には、レート歪み関数 $J=R+\lambda \cdot D$ を用いる。先行研究[8][9]との公平な比較のため、 λ は 0.005 および 0.01 に設定し、本実験では歪み尺度 D として MSE を用いる。学習データセットには ImageNet および CLIC データセットを使用し、 2×10^6 イテレーション学習を行う。

その後、[10]の手法に従い、事前学習済みモデルを量子化し、量子化後のモデルをファインチューニングする。ファインチューニングは合計 4×10^4 イテレーション実施する。テストには Kodak データセット[16]を用いる。なお、以降の節で示す符号化性能は、浮動小数点モデルではなく、すべて整数量子化モデルに基づく結果である。

ハードウェアについて、クロックサイクル単位でハードウェアの挙動を制御できるように、Verilog を用いて実装を行う。合成および配置配線には Vivado を使用し、最終的にビットストリームを生成する。検証に関しては、事前学習済みの量子化 TensorFlow モデルを、ハードウェアと完全に同一の計算プロセスを持つ Python モデルへ変換するコンパイラを作成した。その後、ソフトウェアの実行結果とハードウェアの実行結果を比較することで、機能検証を行う。

2.7 実験結果

ベースラインのネットワークは、分析変換と合成変換にはそれぞれ 5 つの層があり、最初の 4 層のチャンネル数は、分析変換と合成変換でそれぞれ $\{128, 128, 128, 192\}$ と $\{192, 128, 128, 128\}$ である。ニューラルネットワーク探索の結果は以下の通りである。ZC706 では、最適化されたチャンネル数は、分析変換と合成変換でそれぞれ $\{128, 120, 128, 190\}$ と $\{198, 124, 120, 124\}$ である。KU115 では $\{128, 136, 128, 184\}$ と $\{192, 112, 144, 128\}$ である。VCU118 の場合、結果は $\{120, 120, 144, 192\}$ と $\{192, 128, 128, 128\}$ である。なお、VCU118 の合成変換については、探索したネットワークの FPS は元々と同じであるため、ネットワークの調整は行わない。

そして、探索されたネットワークの圧縮性能を表 I で評価する。Kodak データセットにおいて、ZC706 の提案ネットワークと[9]を比較すると、bpp が小さいほど PSNR が悪化している。KU115 と[9]を比較すると、PSNR が向上し、bpp も小さくなっている。VCU118 の状況は ZC706 と同様である。また、明確な比較のために R-D

コストのメトリックも提供する。ZC706 と KU115 の探索されたネットワークは、R-D コストで[9]より優れていることがわかる。VCU118 の探索されたネットワークは、R-D コストで[9]より若干悪い。

表 I. [9] との圧縮性能比較

Network	PSNR (dB) ↑	bpp ↓	R-D cost ↓
[9]	29.94	0.2542	0.6373
ZC706	29.86	0.2434	0.6344
KU115	30.06	0.2531	0.6227
VCU118	29.73	0.2367	0.6424

また、[9] とのハードウェア性能比較を表 II で評価する。きめ細かなパイプライン・アーキテクチャを持つ最新の研究[9]よりも、最大 1.5 倍高速なスループットを達成できることを示している。

表 II. [9] とのハードウェア性能比較

Board	Enc/dec	Throughput of [9]	Throughput of proposal
ZC706	Enc	360P@25.68FPS	360P@33.91FPS
	Dec	360P@28.55FPS	360P@35.00FPS
KU115	Enc	720P@25.04FPS	720P@37.74FPS
	Dec	720P@24.66FPS	720P@30.14FPS
VCU118	Enc	1080P@19.83FPS	1080P@25.53FPS
	Dec	1080P@20.42FPS	1080P@25.53FPS

また、Xilinx DPU によって高速化された最近の研究 [8] と比較して、本研究では、より高い圧縮性能を維持しつつ、より高速なスループットを達成できる。[8] との比較を表 III に示す。[8] と比較すると、エンコードで 1.98 倍、デコードで 8.51 倍のスループットを実現している。また、圧縮性能についても、2 つの圧縮品質、すなわち λ の設定において [8] を上回っている。特に、 $\lambda = 0.01$ の場合には、より高い PSNR とより低い bpp を達成している。

表 III. [8] との圧縮性能およびハードウェア性能比較

Method	Proposal	[8]
Platform	VCU118	ZCU104
Technology	16nm	16nm
Encoding FPS ↑	25.53	12.87
Decoding FPS ↑	25.53	3.00
PSNR ($\lambda=0.005$) ↑	29.73	29.89
bpp ($\lambda=0.005$) ↓	0.2367	0.2565
PSNR ($\lambda=0.01$) ↑	31.75	31.50
bpp ($\lambda=0.01$) ↓	0.4055	0.4165

実際、このアルゴリズムとアーキテクチャの協調最適化の方法論は、他のアプリケーションにも適用できる。今後の課題としては、探索空間を拡大し、異なる層とチャネルを持つより多くのニューラルネットワークを探索する計画である。

3 過学習型 LIC による復号計算量の削減

3.1 研究背景

学習型画像圧縮の分野は近年著しい進展を遂げており、Variational Autoencoder (VAE) に基づく手法は、HEVC などの従来型コーデックを上回るほど競争力の高いレート歪み (R-D) 性能を達成している。しかし、この高い性能は復号計算量の増大を伴い、それにより高遅延および高消費電力が生じるため、リソース制約

のあるデバイスへの展開において大きな障壁となっている。このような計算量と性能のトレードオフに対処するための代替的なパラダイムとして、過学習型ニューラル画像コーデックが登場している。Cool-Chic[11][12][13]、C3[14]、LotteryCodec[15]などのモデルでは、各画像に対して固有の極めて軽量のデコーダを過学習させる。この結果、理論的な復号計算量は VAE ベースの手法と比べて桁違いに低くなり、低消費電力ハードウェアへの実装に適した方式となっている。

3.2 Cool-Chic アーキテクチャ

Cool-Chic は、極めて低い復号計算量を実現するために設計された過学習型コーデックである。VAE ベースのコーデックとは異なり、そのエンコーダは各画像に対して固有の軽量デコーダを学習する。得られたデコーダパラメータは、画像の潜在表現とともにビットストリーム内で伝送される。

復号アーキテクチャは Autoregressive Module (ARM) と、 g_s と表される画像合成経路から構成される。ARM は、各潜在変数について、その因果的近傍に基づいて確率分布、すなわち平均 μ_i およびスケール σ_i を予測することで、エントロピー符号化のためのコンテキストを提供する。

デコーダは量子化された潜在表現 $\hat{y}$ に対して動作するが、この潜在表現は符号化過程に由来する。具体的には、まず画像に対して全精度の潜在表現 y が最適化され、その後量子化される。画像合成経路 g_s は、復号された潜在表現 $\hat{y}$ から最終的な再構成画像 $\hat{x}$ を生成する。この経路は、空間解像度を高める Upsampling モジュールと、アップサンプリングされた特徴を画素値へ写像する最終的な Synthesis モジュールという 2 つの異なる段階から構成される。

これら 3 つのモジュール、すなわち ARM、Upsampling、および Synthesis のパラメータ ($\theta_{\text{arm}}, \theta_{\text{ups}}, \theta_{\text{syn}}$) は、すべて画像ごとに最適化され、ビットストリームに含める必要がある。

3.3 量子化粒度

元の 16 ビット per-block 貪欲探索を 8 ビット重みのパイプラインに適用しても有効ではなく、深刻な性能低下を招くことが確認された。具体的には、BD-Rate オーバーヘッドが +27.52% に達した。この性能低下の要因を分析したところ、異なるモデル層間における動的範囲の大きな差異に加え、単一層内のチャンネル間においても動的範囲に大きなばらつきが存在することが確認された。

このような分布に対して、粗い粒度の量子化は適していない。共有スケールが最大の外れ値によって決定されるため、ゼロ付近に集中している大多数の重みに対して、致命的な精度低下を引き起こすためである。

以上の分析から、ネットワークの忠実度を維持するためには、より細かい粒度、すなわちチャンネルごとの量子化が不可欠であることが示唆される。チャンネルごとの量子化方式では、各層の重みテンソルにおける各出力チャンネルに対して個別のスケール係数を計算する。また、元のコーデックの設計方針との一貫性を保つため、本研究の量子化方式も対称量子化を採用し、スケール係数には 2 のべき乗を用いる。

3.4 パラメータ依存性解析

RTN のような手法の性能上限は、強いパラメータ相関を考慮できない点に起因している。これらの関係を調べるため、本研究では損失ランドスケープの二次特性を解析する。重みに関する損失のヘッセ行列 $H_{ij} =$

$\frac{\partial^2 L}{\partial w_i \partial w_j}$ は、これらの相互作用を詳細に表すものである。

第一に、主対角線上に見られる密なブロックは、強い層内依存性を示しており、単一の層内の重みが高度に相関していることを確認している。第二に、顕著な非対角ブロックは、体系的な層間依存性を示しており、ある層における量子化の選択が、後続の層における誤差ランドスケープを本質的に変化させることを示している。

したがって、各重みを独立に扱う孤立的なパラメータ量子化は最適ではない。このような方法では、相関を持つ量子化誤差がネットワーク全体にわたって累積してしまうためである。

Hessian 解析により強いパラメータ依存性を明らかにし、依存性を考慮した GPTAQ アルゴリズム[18]を適用することで、累積的な量子化誤差を効果的に緩和する。これにより、従来の層単位の PTQ 最適化手法を大幅に上回る性能を達成する。

3.5 誤差補償のための原理的な量子化順序

大規模モデルにおいて、GPTAQ が任意の列処理順序を用いることは、効率を維持するうえで実用上必要である。しかし、この単純化は、Cool-Chic のような小規模かつ高感度なネットワークにおいて重要となる最適化の機会を見落としている。このようなコンパクトなモデルでは、量子化の順序が極めて重要である。

量子化誤差をより効果的に補償するためには、最も感度の高い重みを先に処理すべきである。GPTQ[17]における重み順序に関する近年の研究に着想を得て、本研究では、重要度が単調に低下する原理的な順序に従って重み列を処理することで、この枠組みを過学習型コーデックの文脈に適用する。本研究では、パラメータ感度の代理指標として、ヘッセ行列の対角成分、すなわち $\text{diag}(\hat{\mathbf{X}}^T \hat{\mathbf{X}})$ を用いて、この重要度を測定する。

3.6 実験結果

表 IV の結果は、提案手法の有効性を体系的に検証している。まず、量子化粒度が極めて重要な役割を果たすことを確認できる。単純に RTN を層ごとの量子化 (L) からチャネルごとの量子化 (C) へ変更するだけで、Kodak データセットにおける BD-Rate オーバーヘッドは +16.56% から +8.97% に低減され、細かい粒度の量子化方式が不可欠であることが示された。

しかし、この層間の依存性を考慮しない手法には明確な性能上限が存在する。次に、依存性を考慮した誤差補償の必要性について検証した。高度な補償型アルゴリズムは、この性能上限を突破できることが示された。例えば GPTAQ は、オーバーヘッドを +3.26% まで低減し、単純な手法だけでなく、AdaRound や BRECQ のような高度な層単位の最適化手法をも上回る性能を達成した。これは、相関を持つ量子化誤差を積極的に管理することが、最も重要な要因であることを示している。

最後に、原理的な処理順序の有効性は、提案手法によって実証された。提案手法では、重要度が単調に低下する順序を取り入れることで、最終的に +2.95% の BD-Rate という最良の性能を達成した。本手法は、高い精度と高い計算効率を兼ね備えており、最も有効な解決策であることを示している。

表 IV. Kodak および CLIC データセットにおける post-training quantization の評価結果

Method	Precision	Granularity	Kodak	CLIC
Cool-chic	W16A32	B	–	–
Cool-chic	W8A32	B	27.52	38.50
Cool-chic	W8A32	L	11.30	14.50
RTN	W8A16	L	16.56	21.61
GPTAQ	W8A16	L	6.70	4.71
Proposal	W8A16	L	5.76	4.13
RTN	W8A16	C	8.97	16.06
GPTAQ	W8A16	C	3.26	3.21
Proposal	W8A16	C	2.95	2.78

表 V. Kodak データセットにおける過学習型画像コーデックと VAE ベース画像コーデックの符号化性能および復号計算量の比較

Paradigm	Model	Precision	Decoding MACs/pixel	BD-Rate vs. HEVC (%)
Overfitted	Proposal	W8A16	1.12k	-12.63
VAE	QLIC [19]	W8AFP32	79.20 k	+10.96
VAE	Joint [3]	FP32	96.76 k	-11.76
VAE	ELIC [5]	FP32	331.57 k	-26.89

表 V では、本研究の結果を VAE ベースのパラダイムと比較し、復号計算量と符号化性能の間に存在する本質的なトレードオフを明らかにしている。Joint[3]については、より小規模なモデル構成を用いた結果を示している。QLIC[19]は、8ビット重みおよび FP32 活性化、すなわち W8AFP32 を用いながら、量子化誤差を緩和するためにチャネル分割を採用している。

[19]と比較して、本手法はより低い復号計算量で、より高い符号化性能を達成している。復号計算量が

1.12k MACs/pixel である本研究の8ビットモデルは、VAE ベースの手法 [3], [5] と比べて70倍以上低い計算量を実現している。また、本手法は[3]と同程度の符号化性能を達成している。[5] との符号化効率の差については、モデルサイズの増加などのアプローチにより、今後さらに縮小できる可能性があり、今後の研究で検討する予定である。

4 可変ビットレート LIC

4.1 研究背景

LIC では通常、最適なレート歪み性能を得るために、複数のモデルを個別に学習する必要があり、時間および計算資源の消費が増大する。

近年の画像圧縮研究では、可変ビットレートを実現するために、潜在表現の柔軟性および有効性を高めることに注目が集まっている。その代表的な手法の一つが gain unit の設計であり、これは潜在表現の特定の性質を活用することで、柔軟なスケールリングを可能にする。

これらのユニットは通常、gain 行列 $M \in R^{c \times n}$ から構成される。ここで、 c はチャネル数を表し、 n は異なる gain ベクトルの数を表す。行列 M 内の各 gain ベクトル $m_s = \{m_{s,0}, m_{s,1}, \dots, m_{s,c-1}\}$ は各チャネルに対応しており、スケールリング係数を細かく制御することを可能にする。

この設計により、潜在表現内の各チャネルに固有のスケール値 $m_{s,i} \in R$ を割り当てることができ、各チャネルの重要度に基づいて再構成品質を最適化できる。本手法は、潜在表現におけるチャネル間の不均一な冗長性には対処できるものの、エンコーダおよびデコーダ内部表現間の冗長性については考慮していない。

4.2 マルチゲインユニット

Gain unit は、先行研究[20]において優れた性能を示している。この手法は、潜在表現内に存在する不均一なチャネル冗長性を低減することを目的として設計されている。しかし、このような冗長性は潜在表現だけでなく、エンコーダおよびデコーダの中間表現にも存在すると考えられる。

本研究では、エンコーダが k 個のモジュールから構成され、デコーダが l 個のモジュールから構成されると仮定する。通常、 k と l は等しい。エンコーダ側における処理フローは以下の通りである。

ハイパープライアネットワークについても同様の処理を適用し、量子化前のハイパー潜在表現 z に対して gain unit を用いる。また、ハイパーデコーダの前段には inverse gain unit を導入する。一部の研究では、より正確な潜在分布を推定するために、エントロピーモデリングにおいて局所コンテキストモデルを導入している。しかし、本研究ではこの部分には gain unit を適用しない。

デコーダ側についても、エンコーダ側と同様の流れで処理を行う。

4.3 重み付き確率割り当て

一部の手法[21], [22] では、特定のラグランジュ乗数、例えば最大または最小のラグランジュ乗数に基づいてモデルを学習し、その後、他のラグランジュ乗数に対してモデルを再学習する、あるいは段階的な学習戦略を用いている。しかし、実験結果から、このような手法は特定のビットレート条件では優れた性能を示す一方で、他のビットレートでは性能低下が生じることが分かった。

これらの知見に基づき、本研究では、 λ の値に対して異なる重みを割り当てることで、学習中の選択確率を調整する手法を提案する。この手法には主に2つの利点がある。

1 つ目は、学習の複雑さを低減できる点である。重み付き確率割り当てを用いることで、段階的な学習戦略やモデルの再学習を行うことなく、可変ビットレートを実現できる。

2 つ目は、さまざまな VAE ベースのモデルに対して、異なるビットレート条件間で柔軟に性能を調整できる点である。これにより、全体的な性能を最適化することが可能となる。

4.4 実験結果

本研究では、TCM [6] および Ballé 2018 [2] に基づき、multi-gain unit と single gain unit の性能を比較評価した。Kodak データセットを用いた BD-Rate 比較により、single gain unit [20] は、Ballé 2018 および TCM の固定レートモデルと比較して、それぞれ 6.926% および 3.477% のビットレート増加を示した。これに対し、multi-gain unit は、Ballé 2018 および TCM に対して、それぞれ 2.726% および

2. 683% の改善を示した。

5 LIC に対する敵対的攻撃と敵対的学習

5.1 研究背景

多くのニューラルネットワークベースの手法と同様に、LIC も敵対的攻撃に対して脆弱である。敵対的攻撃の影響を軽減するためには、通常、ネットワークをファインチューニングする敵対的学習が用いられる。

本研究では、入力摂動の有無に応じて 2 種類の符号化設定を考える。通常の符号化設定では、原画像 x をエンコーダに入力し、圧縮表現へ変換した後、デコーダによって再構成画像 $\hat{x}$ を生成する。一方、敵対的攻撃の設定では、原画像 x に敵対的ノイズを加えることで、摂動を含む入力画像 x' を生成する。この摂動画像 x' をエンコーダに入力し、デコーダを通して対応する再構成画像 $\hat{x}'$ を得る。これにより、入力画像に加えられた微小な敵対的摂動が、符号化および復号後の再構成結果にどのような影響を与えるかを評価する。この設定は、LIC モデルが入力摂動に対してどの程度脆弱であるかを検証するための基本的な攻撃シナリオである。

Zhu ら[23]は、攻撃手法、モデル、ならびに画像品質やビットレートといった攻撃対象を含む、さまざまな観点から攻撃の有効性を評価している。Chen ら[24]は、複数の攻撃手法を用いて、再構成画像の品質に着目した攻撃を行っている。一方、Liu ら[25]は、複数のモデルにおいて再構成画像のビットレートを攻撃対象としている。しかし、これらの研究はいずれも、LIC モデルにおける攻撃および防御のために異なる損失関数を用いることについては検討しておらず、損失関数の選択は論文ごとに異なっている。

5.2 提案攻撃手法

画像圧縮モデルに対する敵対的攻撃では、損失関数の選択が攻撃目的を定義する上で中心的な役割を果たす。損失関数の形式に応じて、再構成画像の視覚品質（例えば PSNR）や圧縮データのビットレートなど、システムの異なる側面を攻撃対象とすることが可能である。

本研究ではまず、再構成画像の視覚品質を低下させることを目的とした攻撃に焦点を当てる。この劣化を引き起こすために、歪み攻撃で一般的に用いられる平均二乗誤差（Mean Squared Error; MSE）損失関数を採用する。MSE 損失は二つの画像間の距離に基づいて計算され、比較する画像の組み合わせに応じて、異なる定式化を適用することができる。

具体的には、原画像 x 、敵対的画像 x' 、および再構成画像 $\hat{x}$ 、 $\hat{x}'$ の組み合わせに基づき、以下の 3 種類の歪み損失を定義する。

$$\text{Loss}_1 = \text{MSE}(x, \hat{x}')$$

$$\text{Loss}_2 = \text{MSE}(x', \hat{x}')$$

$$\text{Loss}_3 = \text{MSE}(\hat{x}, \hat{x}')$$

これらの異なる損失定義により、各定式化が攻撃の有効性および対象モデルの頑健性にどのような影響を与えるかを比較分析することが可能となる。

画像品質を対象とした攻撃に加えて、本研究ではビットレートを増加させることを目的とした攻撃についても検討する。これにより、圧縮データのサイズを増大させることができる。ビットレート攻撃では、損失関数を、ハイパープライア構造におけるエントロピーモデルに基づき、潜在表現 y' およびハイパープライア潜在表現 z' の期待負対数尤度として定義する。具体的には、ビットレート損失は以下のように定式化される。

$$\text{Loss}_{\text{bpp}} = \mathbb{E}_{y', z'} [-\log_2 p_{\hat{y}}(y' | \hat{z})] + \mathbb{E}_{z'} [-\log_2 p_z(z')]$$

ここで、 $p_{\hat{y}}$ および p_z は、それぞれ潜在表現 y' および z' に対して学習されたエントロピーモデルを表す。この損失関数を攻撃時に用いることで、圧縮出力のビットレートを増加させ、圧縮効率を低下させることが可能となる。

5.3 提案防御手法

LIC モデルは、外部からの敵対的攻撃に対して脆弱であることが知られている。その対策として、敵対的学習が広く研究されている。この手法では、学習データセット内の敵対的画像を用いて LIC モデルをファインチューニングすることで、このような攻撃に対するロバスト性を向上させる。

しかし、敵対的画像のみを用いてモデルを学習すると、元のモデルと比較して、攻撃を受けていないクリーンな画像に対する性能が大きく低下することが多い。この問題を緩和するため、本研究では、敵対的摂動を加えた画像、すなわち「汚染された」画像と、クリーンな画像の両方を学習データセットに含める。この混合データを用いるアプローチにより、クリーンな画像に対しても高い性能を維持することが可能となる。

本手法では、学習中にモデルの重みを反復的に更新しながら、敵対的画像を生成する。これらの敵対的画像は各学習ステップで更新されたモデルに合わせて生成されるため、より効果的な防御性能が期待される。

5.4 攻撃に関する実験結果

再構成品質を対象とする攻撃において、 $Loss_2$ は一貫して $Loss_1$ および $Loss_3$ よりも有効性が低く、攻撃のための歪み損失関数としては適していないことが示唆される。全体として、[23]で用いられている $Loss_1$ と $Loss_3$ は同程度の性能を示す。

具体的には、 $Loss_1$ は factorized-prior モデル[1]の品質設定 2、4、6、および hyperprior モデル[2]の品質設定 2 において、より有効である。一方、 $Loss_3$ は factorized-prior モデルの品質設定 8、および hyperprior モデルの品質設定 4、6、8 において、 $Loss_1$ を上回る性能を示す。このことから、再構成品質が高くなるにつれて、 $Loss_3$ に基づく攻撃がより有効になることが示唆される。平均 RD cost の観点から見ても、 $Loss_3$ は最も強力な攻撃である。

ビットレートを対象とする攻撃では、モデル品質が高くなるにつれて攻撃効果が増大する。すべての品質設定において、hyperprior モデルは factorized-prior モデルよりも一貫して脆弱である。これは両者の構造的な違いに起因すると考えられる。すなわち、factorized-prior モデルは潜在表現 y のみを用いるのに対し、hyperprior モデルでは追加の潜在表現 z が導入されるため、ビットレート攻撃に対する感度が高くなると考えられる。

さらに、[25]では 600 ステップの FGSM ベースのビットレート攻撃を用いているのに対し、本研究の 40 ステップの PGD ベース攻撃は、品質設定 4 の factorized-prior モデルにおいて、より高い RD cost である 1.658 を達成している。[25]の結果は 1.61 であり、本手法は大幅に少ないイテレーション数でより高い攻撃性能を実現していることを示している。

5.5 防御に関する実験結果

損失関数の中で、 $Loss_3$ に基づく防御は、攻撃側も $Loss_3$ を用いる場合には最も有効である。しかし、他の攻撃タイプに対しては性能が低く、さらにクリーンな入力画像に対する性能も大きく低下する。このことから、 $Loss_3$ は汎用的な防御には適していないことが示唆される。

$Loss_2$ に基づく防御は、ビットレートを対象とする攻撃を除く多くの攻撃に対して中程度の有効性を示し、クリーンな入力画像に対する性能低下は最も小さい。

一方、 $Loss_1$ に基づく防御は、すべての攻撃タイプに対して一貫して高いロバスト性を達成しつつ、クリーンな画像に対する性能低下も比較的小さく抑えている。実際に使用される攻撃手法が不確定であることを考慮すると、幅広い攻撃に対して有効であり、かつクリーン画像の品質を維持できる防御戦略が望ましい。

したがって、 $Loss_1$ を用いた敵対的学習は、最もバランスが取れており、有効な防御手法であると考えられる。

6 まとめ

本研究は、複数の観点から学習型画像圧縮 (LIC) の実用性を向上させるものである。今後の課題としては、これらの手法を一つの統合的な LIC モデルへ組み込むことが挙げられる。

【参考文献】

- [1] J. Ballé, V. Laparra, and E. P. Simoncelli, “End-to-end optimized image compression,” 2016, arXiv:1611.01704.
- [2] J. Ballé, D. Minnen, S. Singh, S. J. Hwang, and N. Johnston, “Variational image compression with a scale hyperprior,” in Proc. Int. Conf. Learn. Represent., 2018, pp. 1–47.
- [3] D. Minnen, J. Balle, and G. D. Toderici, “Joint autoregressive and hierarchical priors for learned image compression,” Advances in neural information processing systems, vol. 31, 2018.
- [4] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, “Learned image compression with discretized gaussian mixture likelihoods and attention modules,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 7939–7948.
- [5] D. He, Z. Yang, W. Peng, R. Ma, H. Qin, and Y. Wang, “Elic: Efficient learned image compression with unevenly grouped spacechannel contextual adaptive coding,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 5718–5727.
- [6] J. Liu, H. Sun, and J. Katto, “Learned image compression with mixed transformer-cnn architectures,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 14388–14397.
- [7] B. Bross et al., “Overview of the versatile video coding (VVC) standard and its applications,” IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 10, pp. 3736–3764, Oct. 2021.
- [8] C. Jia, X. Hang, S. Wang, Y. Wu, S. Ma, and W. Gao, “FPX-NIC: An FPGA-accelerated 4K ultra-high-definition neural video coding system,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 9, pp. 6385–6399, Sep. 2022.
- [9] H. Sun, Q. Yi, F. Lin, L. Yu, J. Katto, and M. Fujita, “F-LIC: FPGA-based learned image compression with a fine-grained pipeline,” in Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC), Nov. 2022, pp. 1–3.
- [10] H. Sun, L. Yu, and J. Katto, “Learned image compression with fixed-point arithmetic,” in Proc. Picture Coding Symp. (PCS), Jun. 2021, pp. 1–5.
- [11] T. Ladune, P. Philippe, F. Henry, G. Clare, and T. Leguay, “Coolchic: Coordinate-based low complexity hierarchical image codec,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 13 515–13 522.
- [12] T. Leguay, T. Ladune, P. Philippe, G. Clare, F. Henry, and O. Deforges, “Low-complexity overfitted neural image codec,” in 2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP). IEEE, 2023, pp. 1–6.
- [13] P. Philippe, T. Ladune, G. Clare, F. Henry, T. Blard, and T. Leguay, “Upsampling improvement for overfitted neural coding,” in 2025 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2025, pp. 1–5.
- [14] H. Kim, M. Bauer, L. Theis, J. R. Schwarz, and E. Dupont, “C3: High-performance and low-complexity neural compression from a single image or video,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 9347–9358.
- [15] H. Wu, G. Chen, P. L. Dragotti, and D. Gunduz, “Lotterycodec: Searching the implicit representation in a random network for low-complexity image compression,” arXiv preprint arXiv:2507.01204, 2025.
- [16] R. Franzen. (1999). Kodak Lossless True Color Image Suite. [Online].Available: <http://r0k.us/graphics/kodak>
- [17] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, “Gptq: Accurate post-training quantization for generative pre-trained transformers,” arXiv preprint arXiv:2210.17323, 2022.
- [18] Y. Li, R. Yin, D. Lee, S. Xiao, and P. Panda, “Gptaq: Efficient finetuning-free quantization for asymmetric calibration,” arXiv preprint arXiv:2504.02692, 2025.
- [19] H. Sun, L. Yu, and J. Katto, “Q-lic: Quantizing learned image compression with channel splitting,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 35, no. 4, pp. 3798–3811, 2025.

- [20] Cui, Ze, et al. "Asymmetric gained deep image compression with continuous rate adaptation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.
- [21] Chen, Tong, and Zhan Ma. "Variable bitrate image compression with quality scaling factors." ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020.
- [22] Tong, Kedeng, et al. "Qvrf: A quantization-error-aware variable rate framework for learned image compression." 2023 IEEE International Conference on Image Processing (ICIP). IEEE, 2023.
- [23] T. Zhu, H. Sun, X. Xiong, X. Zhu, Y. Gong, Y. Fan, et al., "Attack and defense analysis of learned image compression," arXiv preprint arXiv:2401.10345, 2024.
- [24] T. Chen and Z. Ma, "Toward robust neural image compression: Adversarial attack and model finetuning," IEEE Transactions on Circuits and Systems for Video Technology, vol. 33, no. 12, pp. 7842–7856, 2023.
- [25] K. Liu, D. Wu, Y. Wu, et al., "Manipulation attacks on learned image compression," IEEE Transactions on Artificial Intelligence, vol. 5, no. 6, pp. 3083–3097, 2023.

〈発 表 資 料〉

題 名	掲載誌・学会名等	発表年月
Post-Training Quantization for Overfitted Neural Image Coding	IEEE International Symposium on Circuits and Systems (ISCAS)	2026 年 5 月
Efficient Adversarial Attack and Training on Learned Image Compression	APSIPA Annual Summit and Conference	2025 年 10 月
Variable Bitrate Models For Learned Image Compression with Multi-gain units and Weighted Probability Assignment	IEEE International Conference on Visual Communications and Image Processing (VCIP)	2024 年 12 月
FPGA Codec System of Learned Image Compression with Algorithm-Architecture Co-Optimization	IEEE Journal on Emerging and Selected Topics in Circuits and Systems	2024 年 6 月