

CNNを用いたHEVCイントラピクチャのレート制御に関する検討 A Study on Efficient Intra Frame Rate Control Using Convolutional Neural Networks for HEVC

工藤忍[†]
Shinobu Kudo

北原正樹[†]
Masaki Kitahara

清水淳[†]
Atsushi Shimizu

1. はじめに

H.264/AVCやH.265/HEVCなどの映像符号化標準規格では、与えられた符号量に対して最適な符号化処理を行うレート制御技術が重要な役割を持つ。HEVCでは、フレームをCTUと呼ばれるブロックに分割し、ブロック単位で量子化パラメータを制御することでレート制御を行うが、一般的にフレーム内の各ブロックは符号化の難しさが異なるため、各ブロックに対して均一に符号量を割り当ててしまうと画質の低下が生じる。従来はSATDに基づく複雑さを計算し、その複雑さに比例して符号量を割り当てていたが、精度が低いという問題があった。本稿では、CNNを用いたHEVCイントラピクチャのレート制御手法を提案する。提案法では、CNNを用いてフレームの目標符号量に応じてブロック符号量を割り当てることで従来よりも高画質なレート制御を実現する。評価実験により本手法の有効性を示す。

2. 従来のレート制御手法

HEVCに対する代表的なレート制御モデルの1つにR-λモデル[1]があり、このモデルに基づいた様々な改良手法が提案されている[2][3][4]。ここでは、HEVCの参照モデルHMに採用されている[4]の手法について説明する。

R-λモデルは符号量Rとラグランジアン乗数λの関係に基づいたモデルであり、次の式で表される。

$$\lambda = \alpha \left(\frac{C}{R_{target}} \right)^\beta \quad (1)$$

ここで、Cは符号化対象の複雑さ指標、 R_{target} は符号化対象の目標符号量、 α, β はモデルパラメータである。そして、量子化パラメータQPはλを用いて次の式により算出される。

$$QP = 4.2005 \ln(\lambda) + 13.7122 \quad (2)$$

[4]では式(1)及び式(2)をCTU単位で計算し、QPを制御する。また、複雑さ指標Cは8×8ブロック単位のSATDに基づいた指標を用い、 R_{target} はその複雑さ指標に比例した値を割り当てる。

3. 提案法

3.1. 着眼点と提案フロー

従来手法では、SATDに基づく複雑さ指標に比例してブロックの目標符号量を決定しているが、最適な符号量の割り当て比率は符号化対象の複雑さだけではなく、フレームの目標符号量に対しても変動すると考え

[†]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

表1 ネットワーク構成

Type	Kernel	Stride	Outputs (Filters)
Input	-	-	1
Conv.	3 × 3	2 × 2	8
Conv.	3 × 3	2 × 2	16
Conv.	3 × 3	2 × 2	32
Conv.	3 × 3	2 × 2	64
FC	-	-	256
FC (Output)	-	-	52

られるため、フレームの目標符号量に応じてブロックの複雑さ指標を推定できれば、精度向上が期待できる。しかし、そのようなモデルを直接定義することは複雑で困難であるため、本稿では次のようなフローを提案する。まず、CTU毎の各QPに対応する符号量推定器を定義する。次に、推定器の出力を各QPについて全CTU和を計算し、フレーム目標符号量に最も近いQPを得る。そして、そのQPに対応する推定器の出力割合を複雑さ指標として用いる。ここで、符号量推定器は複数の入出力であるため人手による定義は難しい。そこで、本稿では複雑な入出力に対しても対応付けが容易な学習ベースモデルにより実現する。

3.2. 符号量推定器の構成

符号量推定器はCNNベースモデル(表1)で、入力とするi番目のCTUのY成分輝度 $x(i)$ に対して、QP毎のスカラー値 $y(i, QP)$ を出力とする。また、最終層を除く各層の出力には、ReLU及びバッチ正規化を適用する。教師データは各QPでHM[5]のイントラ予測符号化した際の実符号量とし、損失関数には平均自乗誤差を採用する。

3.3. CNNを用いたレート制御手法

複雑さ指標 $C(i)$ を下記のようにCNN出力の各QPの総和がフレーム目標符号量に最も近い値の比率として算出する。ここで、Mはフレーム内のCTU数である。

$$C(i) = \frac{y(i, QP_{init})}{\sum_{j=0}^{M-1} y(j, QP_{init})} \quad (3)$$

$$QP_{init} = \arg \min_{QP} |R_{target} - \sum_{i=0}^{M-1} y(i, QP)| \quad (4)$$

複雑さ指標以外については従来法と同様の処理を実施する。

4. 評価実験

4.1. 実験条件

本提案手法をHEVCの参照モデルであるHM16.6[5]に実装した。表2に符号化条件を示す。テスト画像には

表2 符号化条件

テスト画像	JCT-VC ClassB, SVT (HD), ITE 2nd Ed.
フレーム数	先頭1フレーム
プロファイル	Main 10 Profile, All Intra [8]
目標符号量	3M, 6M, 9M, 12Mbytes

表2に示す計37種類のFullHD画像(先頭フレーム)を用いた。CNNの学習では、データセットはDiv2K[6], パッチサイズは512, 最適化はAdam[7](学習率は 10^{-2} から 10^{-4} までepoch毎に指数減衰)を用い、計20epoch学習した。比較手法にはHM[4]を用いた。

4.2. 結果及び考察

表3に各手法の符号量誤差率(Δ Bits), 提案法の符号化効率率(BD-BR)及び処理時間比率(Δ ET)の平均をそれぞれ示す。表3より, 符号量誤差率は従来法[4], 提案法でそれぞれ0.79%, 0.82%でほぼ同程度となっているが, 提案法は従来法[4]よりも平均で2.16%の符号化効率率ゲインを示した。特にOldTownCrossでは最大の11.56%のゲインが得られた。一方で, BasketballDriveでは4.79%, DucksTakeOffでは2.60%の悪化となってしまった。これらの画像の共通点としてブロックが平坦, 或いはシンプルなテクスチャでイントラ予測の効率が高い点が挙げられる。本手法の学習モデルはイントラ予測の効率まで考慮されていないため, 実際の符号量よりも推定値が大きく出てしまい, 誤った符号量配分をしてしまったと考えられる。そのため, イントラ予測効率も考慮し, CNNの入力に参照画素を含める等を行うことで更なる改善が期待できる可能性がある。また, 符号化処理時間に関しては0.34%とほぼ同程度の処理時間であった。

5. まとめ

本稿ではCNNを用いたHEVCイントラピクチャのレート制御手法を提案した。評価実験により従来よりも平均で2.16%の符号化効率率ゲインを得られることを示した。今後は, 様々な解像度に対する評価実験やインターピクチャへの拡張等を行う予定である。

謝辞

本稿で用いた映像の一部はNTTドコモ様の許諾を得て使用したものです。この場を借りて御礼申し上げます。

参考文献

- [1] M. Jiang and N. Ling, "On Lagrange Multiplier and Quantizer Adjustment for H.264 Frame-layer Video Rate Control," IEEE Trans. Circuits Syst. Video Technol., Vol. 16, No. 5, pp. 663-669, 2006.
- [2] M. Wang and B. Yan, "Lagrangian Multiplier based Joint Three-layer Rate Control for H.264/AVC," IEEE Signal Process. Lett., Vol. 16, No. 8, pp. 679-682, 2009.

表3 符号化性能結果

Sequence	HM [4]		Proposed	
	Δ Bits	Δ Bits	BD-BR	Δ ET
JCT-VC ClassB				
Kimono	0.41%	0.62%	0.93%	1.80%
ParkScene	0.80%	0.98%	-0.90%	-2.92%
Cactus	0.89%	1.28%	0.87%	-3.96%
BasketballDrive	0.64%	0.95%	4.79%	-10.82%
BQTerrace	0.94%	0.78%	-1.00%	-2.98%
SVT				
CrowdRun	0.88%	0.91%	-5.79%	0.79%
DucksTakeOff	0.55%	0.73%	2.60%	0.81%
IntoTree	0.82%	0.93%	-6.46%	0.60%
OldTownCross	0.85%	0.62%	-11.56%	-1.40%
ITE 2nd Ed.				
GinkgoTrees	0.43%	0.44%	-0.84%	1.36%
TruckTrain	0.57%	0.54%	-3.85%	0.83%
CosmosFlowers	0.38%	0.41%	0.69%	1.76%
RedLeaves(PanUp)	0.55%	0.54%	1.01%	1.96%
SunlightThroughLeaves	1.12%	1.08%	-0.02%	3.16%
RedLeaves(PanDown)	0.95%	0.99%	0.08%	1.94%
WomanAtHarbor(CircleDolly)	0.77%	0.63%	-9.75%	-2.52%
Fountain(Follow)	0.80%	0.82%	-4.37%	0.40%
Fountain(Dolly)	0.64%	0.70%	-1.71%	0.86%
StudioConcert(Confetti)	1.50%	1.72%	-0.26%	1.19%
StudioConcert	1.19%	1.24%	-0.42%	1.39%
DramaSet(Day)	0.60%	0.73%	0.02%	1.46%
DramaSet(Night)	0.76%	0.77%	-0.73%	1.41%
Basketball	0.82%	0.74%	-1.06%	1.13%
TwilightScene(ZoomOut)	1.21%	1.01%	-7.37%	1.13%
TwilightScene(Fixed)	0.90%	0.76%	-7.19%	-0.44%
NightScene(Fixed)	0.98%	0.83%	-10.11%	1.49%
HorseRacing(Dirt)	0.43%	0.43%	-1.42%	0.13%
RotatingDisk59.94P	0.29%	0.28%	-2.36%	1.49%
RotatingDisk59.94P_2	0.29%	0.25%	-3.76%	1.53%
Pendulum59.94P	1.02%	1.19%	1.14%	-0.41%
Pendulum59.94P_2	0.90%	0.98%	1.14%	1.71%
ColorfulWorldA	1.72%	1.67%	0.72%	1.51%
ColorfulWorldB	1.38%	1.66%	2.30%	0.68%
BlackDress	0.48%	0.29%	-0.06%	1.62%
WhiteDress	0.92%	0.87%	-8.36%	0.77%
FlashPhotography	0.64%	0.58%	-5.75%	1.36%
Fountain(Chromakey)	0.30%	0.36%	-1.27%	1.65%
Overall	0.79%	0.82%	-2.16%	0.34%

- [3] B. Li, H. Li, L. Li and J. Zhang, "Rate control by R-lambda model for HEVC," JCT-VC K0103, 2012.
- [4] X. Wang, L. Yu and J. Chen, "Intra frame rate control based on SATD," JCT-VC M0257, 2013.
- [5] HM software 16.6, <https://hevc.hhi.fraunhofer.de/trac/hevc/browser/branches/HM-16.6-dev>.
- [6] E. Agustsson and R. Timofte, "NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study," IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, July 2017.
- [7] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6948, 2014.
- [8] F. Bossen, "Common HM test conditions and software reference configurations," JCT-VC L1100, 2013.