ANÁLISE DO IMPACTO DA PRECISÃO DE QUANTIZAÇÃO NO DECODIFICADOR DE HYPERPRIOR DO SSF

RUHAN CONCEIÇÃO[1]; WEN-HSIAO PENG[2]; MARCELO PORTO[3]; LUCIANO AGOSTINI[4]

[1]*Universidade Federal de Pelotas – radconceicao@inf.ufpel.edu.br*
[2]*National Yang Ming Chiao Tung University – wpeng@cs.nctu.edu.tw*
[3]*Universidade Federal de Pelotas – porto@inf.ufpel.edu.br*
[4]*Universidade Federal de Pelotas – agostini@inf.ufpel.edu.br*

1. INTRODUÇÃO

Nos últimos anos, codecs neurais surgiram como uma alternativa promissora aos métodos tradicionais de codificação de vídeo, explorando arquiteturas baseadas em autoencoders e modelos probabilísticos aprendidos. Em particular, a introdução do hyperprior (BALLE, 2018) representou um avanço significativo, pois permite modelar de forma mais precisa as distribuições do código latente e, consequentemente, reduzir a entropia da informação transmitida. Esse recurso foi incorporado em arquiteturas modernas como o SSF (AGUSTSSON et al., 2020), que atingem desempenho comparável aos codecs convencionais em diversas condições. No entanto, apesar desse progresso em termos de eficiência de compressão, a implementação prática desses sistemas ainda enfrenta desafios relacionados ao custo computacional, ao consumo de energia e às inconsistências entre plataformas, especialmente no módulo de hyperprior, que é altamente sensível a erros de quantização.

A implantação prática desses modelos enfrenta, portanto, desafios importantes. Em primeiro lugar, a maior parte dos frameworks de compressão neural é treinada e avaliada em floating point (FP32/FP16), o que implica elevado consumo de memória e energia, limitando o uso em dispositivos embarcados. Além disso, diferenças de arredondamento entre implementações de ponto flutuante em CPU, GPU ou DSP podem gerar inconsistências entre plataformas, comprometendo a reprodutibilidade e até mesmo a decodificação correta (BALLE et al., 2019; CONCEIÇÃO et al., 2025).

Uma solução viável para esses desafios é a quantização pós-treinamento (post-training quantization — PTQ), que converte pesos e ativações de modelos neurais para representações inteiras de menor precisão (PYTORCH, 2021). Essa técnica reduz custos de armazenamento e processamento, além de garantir aritmética determinística, eliminando diferenças entre plataformas. Ferramentas como a AIMET (QUALCOMM, 2021) facilitam esse processo, oferecendo estratégias de calibração baseadas em amostras de dados.

Neste resumo expandido, analisamos os impactos da quantização no decodificador de hyperprior do codec SSF, reconhecido como um dos módulos mais sensíveis a erros numéricos. Para isso, consideramos quatro cenários distintos de precisão, nos quais pesos e ativações compartilham o mesmo número de bits: W4A4, W8A8, W12A12 e W16A16. Em cada configuração, o número N em W*N*A*N* indica a quantidade de bits inteiros utilizada para representar os pesos (*weights*) e ativações (*activations*).

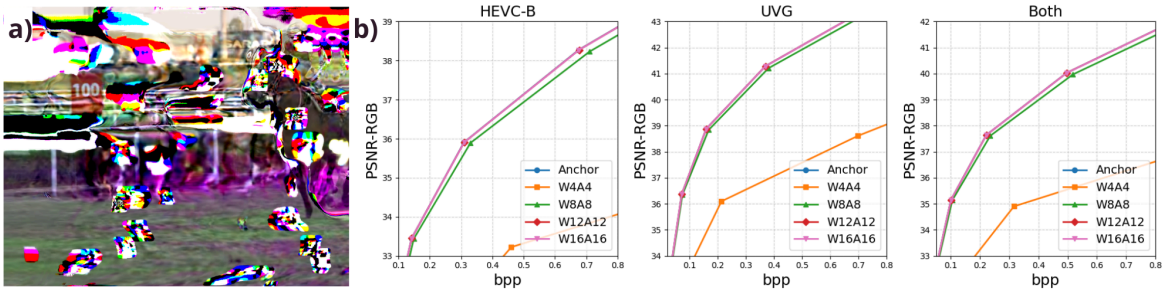

Figura 1. a) Exemplo de resíduo gerado na reconstrução da imagem devido às inconsistências entre plataformas. b) Gráficos de taxa de bits vs. qualidade para o trabalho desenvolvido.

2. METODOLOGIA

Neste trabalho, adota-se o codec SSF, implementado na biblioteca CompressAI (BÉGAUD et al., 2020), e quantizado empregando a biblioteca AIMET (QUALCOMM, 2021). A calibração dos parâmetros de quantização foi realizada com 1000 sequências do conjunto Vimeo90k, recortadas para a dimensão de 256 × 256 pixels, utilizando um modo de definição dos parâmetros de quantização orientado pela métrica Mean Squared Error (MSE). Foram avaliados quatro cenários distintos: W4A4, W8A8, W12A12 e W16A16.

Os testes foram conduzidos com os datasets de vídeo HEVC-B (BOSSEN et al., 2013) e UVG (MERCAT et al., 2020), ajustados para 1920 × 1024 pixels, sendo codificados em sequências de 96 quadros. A codificação utilizou um quadro intra inicial seguido de P-frames. O desempenho em eficiência de compressão foi avaliado através da métrica BD-rate (BJØNTEGAARD, 2001), que estima a variação percentual média na taxa de bits necessária para alcançar a mesma qualidade em curvas RD, permitindo assim comparar diferentes configurações de forma objetiva. Para garantir estabilidade, apenas os níveis de qualidade ímpares (1, 3, 5, 7 e 9) foram considerados, utilizando FP32 como âncora.

3. RESULTADOS E DISCUSSÃO

Tabela 1: BD-rate considerando diferentes precisões. FP32 como âncora.

W4A4	W8A8	W12A12	W16A16
254,0%	5,7%	0,2%	0,1%

A Tabela 1 apresenta os valores médios de BD-rate para os cenários avaliados, obtidos a partir das curvas de taxa-distorção mostradas na Figura 2(b). Observa-se que a perda de eficiência cresce à medida que a precisão é reduzida, embora de forma não linear. Enquanto a configuração W8A8 mantém desempenho competitivo, o caso W4A4 mostra-se inviável.

Os resultados evidenciam que a quantização extrema em W4A4 degrada severamente a eficiência de compressão, resultando em aumento de BD-rate superior a 250%, o que inviabiliza sua adoção. Por outro lado, a configuração W8A8 mostrou-se um ponto de equilíbrio, com degradação limitada a até 7%, sendo adequada para aplicações em tempo real e sistemas embarcados (CONCEIÇÃO et al., 2025). A precisão W12A12 apresentou desempenho praticamente indistinguível do FP32, com perdas médias de apenas 0,2%,

enquanto W16A16 não trouxe ganhos adicionais relevantes em relação ao W12A12, podendo ser considerado redundante.

A análise confirma que o decodificador de hyperprior é altamente sensível a quantizações agressivas, com perdas significativas observadas no caso W4A4. Por outro lado, as configurações W8A8 e W12A12 mostraram-se adequadas para manter a eficiência de compressão mesmo em cadeias longas de 96 quadros, indicando que não há acúmulo relevante de erro ao longo da sequência.

Do ponto de vista prático, o uso de W8A8 oferece uma solução eficiente em termos de custo computacional e consumo de energia, aproveitando instruções otimizadas para inteiros em hardware moderno. Já a configuração W12A12 garante fidelidade quase total ao modelo FP32, sendo recomendada para cenários que exigem alta qualidade ou aplicações críticas.

Adicionalmente, a quantização do decodificador de hyperprior contribui para eliminar erros decorrentes de inconsistências de arredondamento entre plataformas (cross-platform round-off errors) (BALLE et al., 2019; CONCEIÇÃO et al.,, 2025), garantindo maior robustez na sincronização entre codificador e decodificador em implementações heterogêneas.

4. CONCLUSÕES

Este estudo, focado na precisão do decodificador de hyperprior do SSF, permitiu destacar os seguintes pontos principais:

- W4A4 mostrou-se inviável, com aumento de BD-rate superior a 200%.
- W8A8 representa o melhor compromisso entre eficiência e robustez.
- W12A12 elimina quase totalmente a diferença em relação ao FP32, com perdas de até 0,2%.
- W16A16 não apresenta ganhos práticos adicionais.

Dessa forma, conclui-se que, para implantações reais, recomenda-se o uso de W8A8 quando a eficiência for prioritária e de W12A12 quando a fidelidade for imprescindível. A quantização, além de resolver inconsistências entre plataformas, reduz custos de memória e processamento, tornando a adoção de codecs neurais mais viável em dispositivos com recursos restritos.

5. REFERÊNCIAS BIBLIOGRÁFICAS

AGUSTSSON, E.; MINNEN, D.; JOHNSTON, N.; BALLÉ, J.; HWANG, S. J.; TODERICI, G. Scale-space flow for end-to-end optimized video compression. In: IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 2020, Seattle. Proceedings [...]. IEEE, 2020.

BALLÉ, J.; JOHNSTON, N.; MINNEN, D. Integer networks for data compression with latent-variable models. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS (ICLR), 2019, New Orleans. Proceedings [...]. ICLR, 2019.

BALLÉ, J.; MINNEN, D.; SINGH, S.; HWANG, S. J.; JOHNSTON, N. Variational image compression with a scale hyperprior. arXiv:1802.01436, 2018.

BÉGAINT, F.; MENTZER, F.; AGUSTSSON, E.; VAN GOOL, L. CompressAI: a PyTorch library and evaluation platform for end-to-end compression research. arXiv:2011.03029, 2020. Disponível em: https://arxiv.org/abs/2011.03029
. Acesso em: 29 ago. 2025.

BJØNTEGAARD, G. Calculation of average PSNR differences between RD-curves. ITU-T VCEG-M33, 2001.

BOSSEN, F. et al. Common test conditions and software reference configurations. JCTVC-L1100, v. 12, n. 7, 2013.

CONCEIÇÃO, R.; PORTO, M.; PENG, W.-H.; AGOSTINI, L. Cross-platform neural video coding: a case study. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (ISCAS), 2025. Proceedings [...]. IEEE, 2025. p. 1–5.

MERCAT, J.; VIITANEN, V.; VANNE, J. UVG dataset: 50/120fps 4K sequences for video codec analysis and development. In: ACM MULTIMEDIA SYSTEMS CONFERENCE (MMSys), 2020, Istanbul. Proceedings [...]. ACM, 2020. p. 297–302.

PYTORCH. Quantization. 2024. Disponível em: https://pytorch.org/docs/stable/quantization.html
. Acesso em: 29 ago. 2025.

QUALCOMM INNOVATION CENTER, INC. AI Model Efficiency Toolkit (AIMET). 2024. Disponível em: https://github.com/quic/aimet
. Acesso em: 12 ago. 2025.