Compressão de Modelos de Aprendizagem Profunda Aplicada à Esteganálise de Imagens Digitais

Gabriel de Assunção Ferreira
Escola Politécnica de Pernambuco
Universidade de Pernambuco
Recife, Brasil
gaf@poli.br

Francisco Madeiro
Escola Politécnica de Pernambuco
Universidade de Pernambuco
Recife, Brasil
madeiro@poli.br

Verusca Severo de Lima
Escola Politécnica de Pernambuco
Universidade de Pernambuco
Recife, Brasil
verusca.severo@poli.br

***Resumo*—A esteganografia e a esteganálise têm papel fundamental no cenário da segurança da informação. Modelos de aprendizado profundo têm sido aplicados à detecção de esteganografia em imagens digitais. Esses modelos podem ser constituídos de milhares ou milhões de parâmetros, o que pode ser um desafio para sua implementação em dispositivos de recursos limitados, razão pela qual neste trabalho investigam-se técnicas de compressão de um modelo de aprendizado profundo aplicado à esteganálise de imagens digitais. Técnicas de poda, quantização e compartilhamento de pesos são utilizados para um modelo de aprendizado profundo para esteganálise de imagens digitais. Os modelos comprimidos foram avaliados em termos de acurácia e de espaço em memória ocupado. Diferentes estratégias de otimização foram aplicadas para detecção de esteganografia a partir de cada algoritmo considerado. Foi possível obter modelos com acurácias levemente inferiores, iguais ou levemente superiores às apresentadas pelo modelo original e com espaços em memória ocupado reduzidos em até 83%.**

***Palavras-chaves*—esteganálise, aprendizado profundo, compressão, poda, quantização, compartilhamento de pesos.**

I. Introdução

As técnicas de esteganografia introduzem modificações em determinada mídia com o objetivo de embutir uma mensagem não perceptível [1], [2]. Técnicas de esteganografia são comumente empregadas em mídias como, por exemplo, imagens [3], vídeos [4], áudio [5] e texto [6].

A contramedida da esteganografia é denominada de esteganálise. Seu objetivo é detectar a presença de uma informação oculta. Por esse motivo, a esteganálise é considerada como um problema de classificação binária. A acurácia da estratégia empregada é um dos principais aspectos de avaliação [7].

Técnicas de esteganografia são utilizadas para cyberterrorismo [8], vazamento de dados [9], guerra cibernética [10], comunicações militares e disseminação de *malwares* [11], [12]. A esteganálise é empregada, por exemplo, para combate ao terrorismo, rastreamento de atividades criminosas na internet [13], [14] e detecção de *malwares* [15].

Abordagens recentes de esteganálise [16] contam com a utilização de aprendizado de máquina. Em específico, vem sendo comum o emprego de técnicas de aprendizado profundo, mais precisamente com a utilização de redes neurais convolucionais (CNN, do inglês *Convolutional Neural Network*) [17]–[19], cujos resultados em termos de acurácia, em muitos dos casos avaliados, se mostraram superiores aos obtidos a partir da utilização de estratégias clássicas de aprendizado de máquina [16], [20].

As redes neurais artificiais possuem camadas e são compostas por elementos básicos, chamados neurônios, que carregam valores denominados de pesos [21]. As CNNs possuem diversas camadas e parâmetros. As CNNs, em geral, têm o objetivo de resolver problemas de caráter complexos e apresentam estruturas com várias camadas com milhares ou milhões de parâmetros [22]. Essa alta quantidade de elementos implica uma maior necessidade de requisitos de memória para armazenamento dos modelos, principalmente levando em consideração a utilização de dispositivos com recursos limitados.

Diante desse contexto, técnicas de compressão de modelos de aprendizado profundo são propostas com o objetivo de reduzir o espaço em memória ocupado e/ou tempo de resposta do modelo [23]. Entre as principais técnicas de compressão, é válido destacar a poda [24], quantização [25] e compartilhamento de pesos [26]. A utilização das estratégias mencionadas tem como objetivo reduzir os requisitos de espaço em memória ocupado pelo modelo sem que haja um grande impacto em seu desempenho.

Motivado pelo exposto, este trabalho tem como objetivo avaliar a aplicação das técnicas de compressão de modelos de aprendizado profundo, mais especificamente poda, quantização, compartilhamento de pesos e a combinação de poda e quantização, ao modelo GBRAS-Net [27].

O restante do artigo está organizado da seguinte forma: As Seções II e III abordam esteganálise e técnicas de compressão respectivamente. A Seção IV apresenta a metodologia. A Seção V mostra os resultados obtidos. A conclusão é apresentada na Seção VI.

II. Esteganálise em imagens digitais

As técnicas de esteganografia têm como objetivo esconder uma informação ou mensagem em um determinado tipo de mídia como, por exemplo, imagem, áudio ou vídeo. Levando em consideração a ocultação da informação em uma imagem, modificações são realizadas de tal forma que, quando a imagem original (imagem de cobertura) e a modificada (estego-imagem, a qual tem a informação oculta) são dispostas lado

a lado, não há a percepção, a partir da análise visual de um humano, de que se tratam de imagens distintas [28].

A esteganálise tem o papel de detectar a presença de esteganografia. A esteganálise é um problema de classificação binária. Isso significa que existe ou não esteganografia embutida em determinada mídia.

Os métodos de esteganografia e de esteganálise podem ser classificados quanto ao domínio da mídia. Levando em consideração imagens, quando as modificações são realizadas diretamente em seus *pixels*, a esteganografia é realizada no domínio espacial; quando o método de ocultação é realizado a partir de alterações nos coeficientes das transformadas de determinada imagem, a esteganografia é realizada no domínio da frequência [2].

A esteganálise pode ser classificada como universal ou específica. Quando é empregada para detecção da presença de esteganografia independentemente da técnica utilizada para embutir informação, é considerada universal; quando é projetada para detecção de informação oculta a partir de um determinado algoritmo de esteganografia, é classificada como esteganálise específica [2].

Outras abordagens incluem a esteganálise local, que visa detectar a presença de esteganografia em regiões específicas ou subáreas de uma imagem; a esteganálise quantitativa, cujo objetivo é estimar a quantidade de informação oculta; e a esteganálise forense, que busca, entre outros propósitos, identificar o conteúdo embutido na imagem [16].

III. Técnicas de compressão de Modelos de Aprendizado Profundo

A. Poda

As técnicas de poda consistem na eliminação de parâmetros do modelo de aprendizado profundo que podem ser considerados irrelevantes ou redundantes. Uma das principais estratégias de poda consiste na eliminação de conexões com valores abaixo de um valor limite [29], [30].

Uma outra abordagem é a eliminação dos pesos de acordo com uma determinada taxa de poda (TP). A taxa de poda diz respeito à porcentagem de conexões que devem ser eliminadas da rede levando em consideração o seu valor [24].

A Figura 1 apresenta um exemplo da aplicação da estratégia de compressão a partir de poda. Ao lado esquerdo da figura é possível verificar uma rede com diversas conexões entre camadas e neurônios. Após a aplicação da poda, ao lado direito da figura, é possível verificar o modelo com um número de conexões inferior ao apresentado pelo modelo original. A diminuição de parâmetros em uma rede pode significar uma redução no espaço em memória ocupado e um menor tempo de resposta do modelo.

B. Quantização

A quantização é uma técnica de compressão de modelos de aprendizagem profunda e consiste na modificação da representação dos valores dos parâmetros de uma rede. Os valores numéricos em uma rede neural artificial normalmente são representados pelo formato FP32, que possui 32 *bits* com

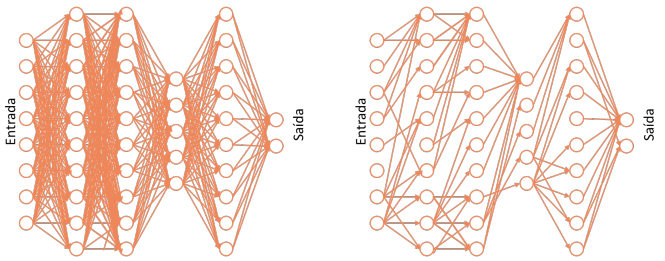

Fig. 1. Aplicação da técnica de poda em uma rede neural artificial.

representação do tipo ponto flutuante (do inglês, *floating-point*) em sua composição. Com a utilização da técnica de quantização, esses valores passam a ser representados com um menor número de *bits* e/ou por números inteiros [25].

A Figura 2 apresenta a quantização de faixa dinâmica de oito *bits* em um bloco 4×4. Os valores são inicialmente representados a partir do formato FP32. Após a quantização, esses valores são representados com 8 *bits* do tipo inteiro. No exemplo em questão, todos os valores são divididos por três e aproximados para o valor inteiro mais próximo. Essa operação acontece para que os valores fiquem dentro de uma faixa de valores possíveis para oito *bits*.

125,43	320,20	-281,90	-120,32
83,49	15,43	-150,30	208,38
-25,89	110,30	80,70	-145,65
-80,23	70,56	3,31	-3,98

→

42	107	-94	-40
28	5	-50	69
-8	37	27	-49
-27	24	1	-1

Fig. 2. Exemplo de estratégia de quantização de faixa dinâmica.

C. Compartilhamento de pesos

Compartilhamento de pesos ou *clusterização* de pesos é uma técnica de compressão de aprendizado profundo que consiste na utilização de um mesmo valor para vários pesos que serão agrupados em uma rede neural. Os pesos mencionados inicialmente são representados por valores diferentes e, após o agrupamento, compartilham do mesmo valor, diminuindo o número de valores distintos no modelo [26].

A Figura 3 apresenta um exemplo de compartilhamento de pesos em um bloco de tamanho 4×4. O número de *clusters* para o exemplo é quatro. Ao lado esquerdo da figura, visualiza-se o bloco de imagens antes da aplicação da técnica de compressão. No caso referido, existem valores variados. Os *pixels* são agrupados em quatro diferentes grupos representados por cores distintas na figura. Cada grupo será representado por um valor de referência, que, no exemplo, é definido pela média dos valores de um grupo. Considerando o grupo de cores verdes, a média dos pesos é -10, representado pelo valor de referência 1. Por fim, ao aplicar a técnica de compartilhamento

de pesos, cada elemento com a cor verde apresenta o valor -10. O mesmo acontece para cada *cluster* representado em outras cores. É possível verificar o bloco resultante com os valores *clusterizados* ao lado direito da Figura 3

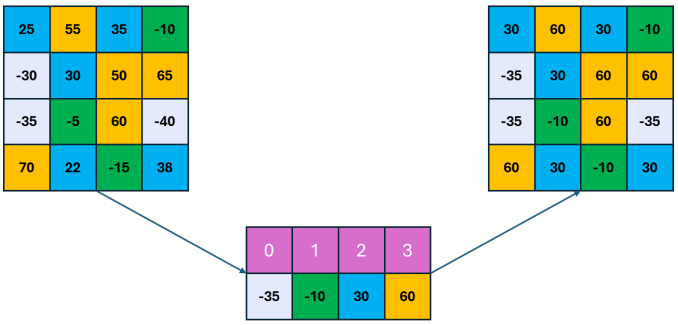

Fig. 3. Exemplo de aplicação da técnica de compartilhamento de pesos para um bloco 4×4 com número de clusters igual a 4.

Um importante fator da técnica de compartilhamento de pesos é o método de inicialização do centróide. Entre alguns métodos de inicialização, é válido destacar a inicialização baseada em densidade [31], k-means++ [32], linear e aleatória.

IV. Metodologia

O modelo de aprendizado profundo GBRAS-Net [27] foi selecionado para aplicação das técnicas de compressão e posterior avaliação do modelo comprimido em termos de acurácia e de espaço em memória ocupado.

A base de dados selecionada para as simulações foi a BOSSBase 1.01, a qual conta com 10.000 imagens distintas em escala de cinza. A base também foi utilizada no trabalho de Reinel *et al.* [27] para avaliação do desempenho do modelo original. Para cada simulação, além das 10.000 imagens originais, há o acréscimo das mesmas 10.000 imagens com informações ocultas a partir de algoritmo de esteganografia, totalizando 20.000 imagens por conjunto de simulação. O conjunto de dados foi distribuído em 4.000 pares de imagens para o treinamento, 1.000 pares para a validação e 5.000 pares para o teste.

Os algoritmos de esteganografia utilizados para avaliação do desempenho das estratégias foram o MiPOD (*Minimizing the Power of Optimal Detector*) [33], HUGO (*Highly Undetectable SteGO*) [34] e HILL (*HIgh-pass, Low-pass and Low-pass*) [35] para taxas de ocultação de 0,2 bpp (*bits* por *pixel*) e 0,4 bpp.

Para cada cenário de detecção da presença de esteganografia a partir de algoritmo e taxa de ocultação específicos, foram aplicadas as técnicas de compressão por meio de quantização, poda e compartilhamento de pesos. Em relação às técnicas de quantização, foram aplicadas a quantização de faixa dinâmica de 8 *bits* (DRQ, do inglês *dynamic range quantization*) e quantização FP16. Considerando a técnica de poda, foi avaliada a aplicação de taxas de poda de 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% ao modelo original. A aplicação da técnica de compartilhamento de pesos foi avaliada com diferentes estratégias de inicialização do centróide, são elas a inicialização baseada em densidade, k-means++, linear e aleatória. Por fim, foram realizadas combinações entre as técnicas de quantização e poda.

Para a realização das simulações, etapas de treinamento, validação e testes dos modelos originais e comprimidos foi utilizado o serviço em nuvem do Google Colab Pro. A máquina utilizada conta com as seguintes configurações: Ubuntu 22.04.3 LTS, Intel(R) Xeon(R) CPU @ 2.20GHz com RAM de 51 GB e GPU NVIDIA A100-SXM4 com uma memória RAM de 40 GB.

Python 3.10 foi a linguagem de programação utilizada para a realização das simulações com a utilização das bibliotecas Tensorflow 2.17.1, Keras 3.5.0 e Tensorflow Model Optimization Toolkit (TFMOT), na versão 0.8.0.

V. Resultados

Essa seção apresenta os resultados obtidos em termos de acurácia e espaço em memória ocupado pelo modelo de aprendizado profundo comprimido. As Figuras 4, 5, 6, 7, 8 e 9 apresentam um gráfico de barras com eixo duplo com o objetivo de comparar os resultados obtidos pelas estratégias empregadas de cada simulação. O eixo vertical posicionado à esquerda apresenta os valores de acurácia, representado em laranja, enquanto o eixo vertical do lado direito apresenta os valores de espaço em memória ocupado em *megabytes* (MB), representado em azul, do modelo de aprendizado profundo.

A. MiPOD

As Figuras 4 e 5 mostram os resultados de acurácia e de espaço em memória ocupado para diferentes estratégias de compressão do modelo original para detecção de informações ocultas a partir do algoritmo MiPOD a uma taxa de ocultação de 0,4 bpp e 0,2 bpp respectivamente.

Considerando o cenário no qual a taxa de ocultação foi de 0,4 bpp, oito modelos com a aplicação de diferentes técnicas de compressão obtiveram acurácias com diferença inferior a 1% em relação à acurácia do modelo GBRAS-Net.

Destacam-se os resultados ao aplicar taxas de podas de 30% e 40%. Para o primeiro cenário mencionado, a perda de acurácia foi de 0,06% em relação ao modelo original com um espaço em memória ocupado reduzido de 0,65 MB para 0,50 MB. Para o segundo caso, a perda de acurácia foi de 0,05% e uma redução do espaço em memória ocupado pelo modelo de 0,65 MB para 0,45 MB. A utilização da combinação da quantização FP16 com uma taxa de poda igual a 60% resultou em uma maior compressão do modelo de aprendizado profundo, com requisitos de memória aproximadamente 70% inferiores aos apresentados pelo modelo original e uma acurácia apenas 0,30% inferior.

Com a aplicação da quantização de faixa dinâmica, foi possível obter um modelo com espaço em memória ocupado reduzido em aproximadamente 75% quando comparado ao modelo original e uma queda de acurácia de apenas 0,48%. Ao combinar a quantização de faixa dinâmica com uma taxa de

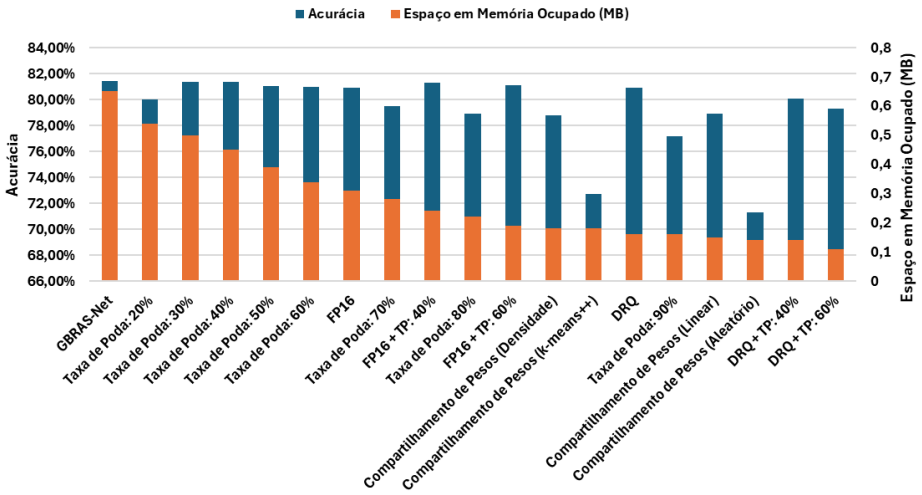

Fig. 4. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso da técnica MiPOD a uma taxa de 0,4 bpp.

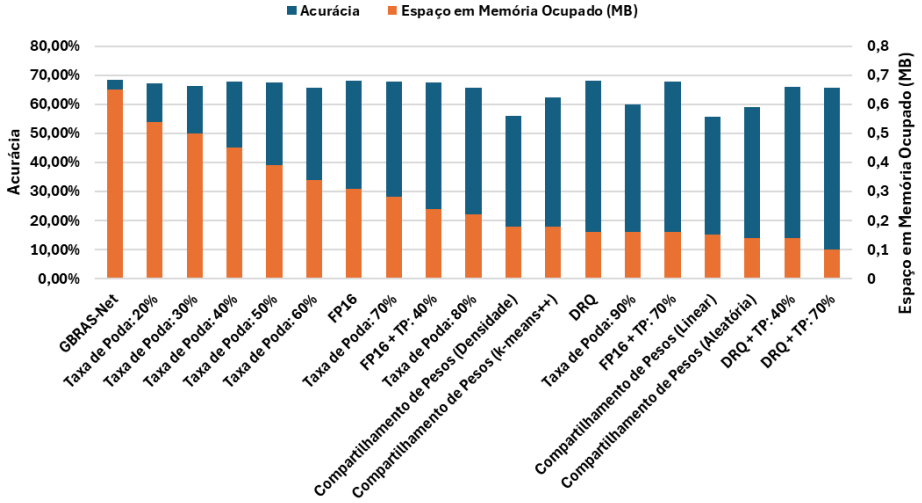

Fig. 5. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso do algoritmo MiPOD a uma taxa de 0,2 bpp.

poda de 40%, o modelo comprimido apresentou um espaço em memória ocupado de 0,14 MB, o que representa uma redução de aproximadamente 78% e uma acurácia 1,30% abaixo da apresentada pelo modelo original.

Em relação às simulações envolvendo o cenário no qual a taxa de ocultação foi de 0,2 bpp, destaca-se o valor obtido pelo modelo quantizado a partir da utilização da quantização de faixa dinâmica. No caso mencionado, o espaço em memória ocupado pelo modelo foi reduzido de 0,65 MB para 0,16 MB quando comparado ao modelo original, enquanto a acurácia apresentada foi apenas 0,24% abaixo da apresentada pelo modelo original. Com a combinação de quantização de faixa dinâmica e taxa de poda de 70%, a redução de requisitos de memória foi de aproximadamente 84%, porém os valores de acurácia apresentaram uma queda de 2,53%.

É possível afirmar, a partir da Figura 5, que sete diferentes modelos comprimidos apresentaram diferença de acurácia abaixo de 1% quando comparada à do modelo original.

B. HILL

As Figuras 6 e 7 apresentam os gráficos de barras com os valores de acurácia e espaço em memória ocupado pelo modelo comprimido e os resultados obtidos para cada estratégia para detecção de informações ocultas a partir do algoritmo HILL a uma taxa de 0,4 bpp e 0,2 bpp respectivamente.

Considerando a taxa de ocultação de 0,4 bpp, destacam-se os resultados obtidos pelo modelo comprimido ao aplicar quantização FP16, taxa de poda de 50%, 60% e 70%, compartilhamento de pesos com inicialização do centróide baseada em densidade e quantização de faixa dinâmica. Foi possível obter

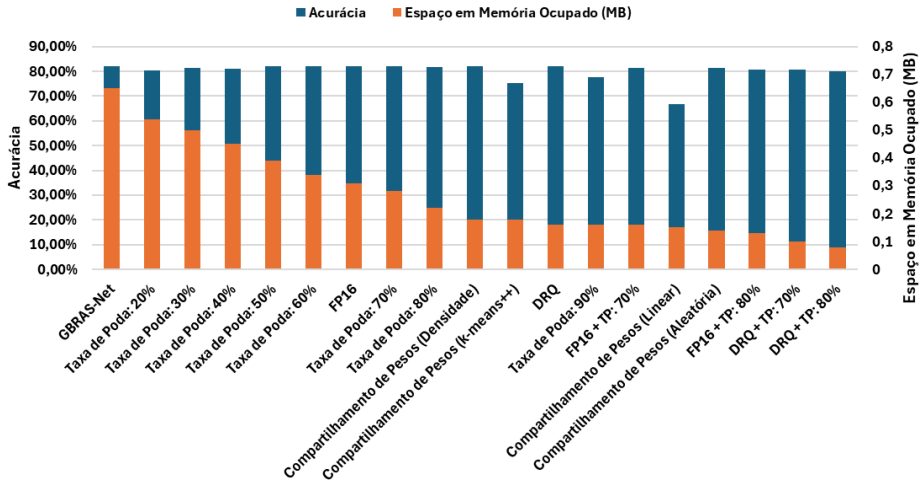

Fig. 6. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso do algoritmo HILL a uma taxa de 0,4 bpp.

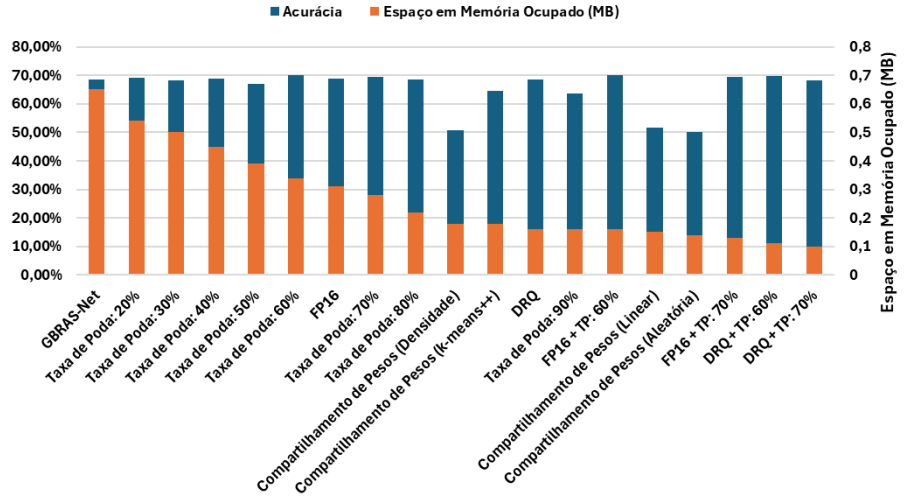

Fig. 7. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso do algoritmo HILL a uma taxa de 0,2 bpp.

valores de acurácia levemente superiores aos apresentados pelo modelo original ao aplicar as estratégias supracitadas.

Os dois melhores resultados em termos de acurácia foram atingidos ao aplicar as estratégias de compartilhamento de pesos com inicialização de centróide baseada em densidade (82,10%) e quantização de faixa dinâmica de 8 *bits* (82,17%). Para o primeiro cenário mencionado, o aumento de acurácia em relação ao modelo original foi de 0,20% e uma redução de espaço em memória ocupado de aproximadamente 72%. Para a segunda estratégia avaliada, o valor de acurácia foi aumentado em 0,27% com uma redução de aproximadamente 75% dos requisitos de memória originais do modelo.

A Figura 7 apresenta os valores obtidos quando o cenário envolve a detecção de informações ocultas a uma taxa de 0,2 bpp. É possível afirmar, a partir da figura, que existem diversos modelos com valores de acurácia semelhantes. Destaca-se o aumento de acurácia de 1,53% obtido pelo modelo comprimido a partir da combinação da quantização FP16 com uma taxa de poda de 60%. Para esse cenário, a redução de espaço em memória ocupado foi de aproximadamente 75%.

Foi possível obter resultados de acurácia levemente superiores ao resultado obtido pelo modelo original com a aplicação da combinação da quantização FP16 e uma taxa de poda de 70% e com a combinação da quantização de faixa dinâmica de oito *bits* e uma taxa de poda de 60%. Para os cenários supramencionados, o espaço em memória ocupado foi reduzido em aproximadamente 80% e 83% respectivamente.

C. HUGO

As Figuras 8 e 9 apresentam os gráficos de barras com valores de acurácia e espaço em memória ocupado para

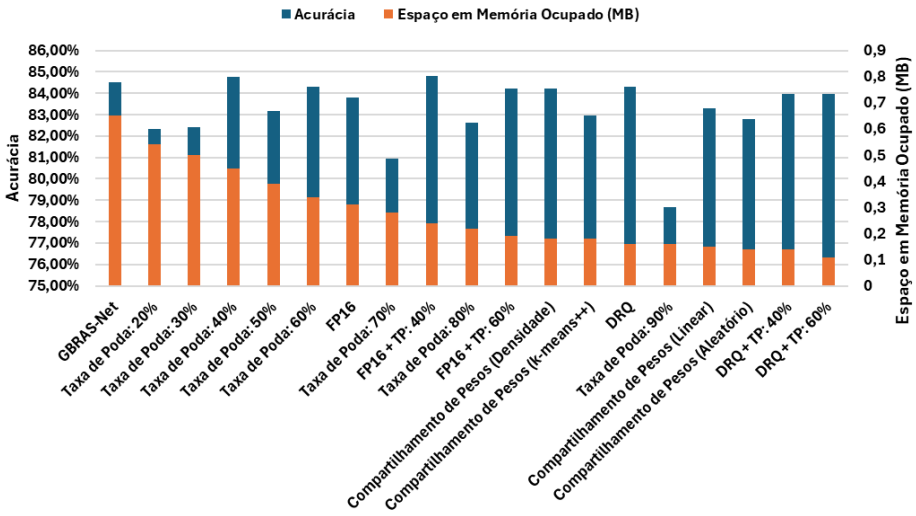

Fig. 8. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso do algoritmo HUGO a uma taxa de 0,4 bpp.

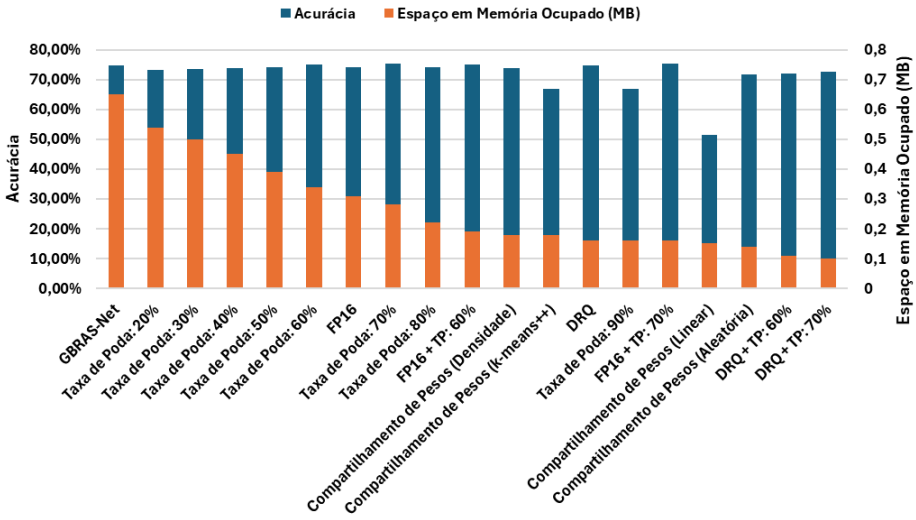

Fig. 9. Resultados de acurácia e espaço em memória ocupado pelo modelo comprimido para diferentes técnicas de compressão para detecção de esteganografia a partir do uso do algoritmo HUGO a uma taxa de 0,2 bpp.

detecção do algoritmo HUGO para taxas de ocultação de 0,4 bpp e 0,2 bpp.

Para uma taxa de ocultação de 0,4 bpp, é possível afirmar que dois modelos com resultados de acurácia levemente superiores ao apresentado pelo modelo original foram obtidos. Ao aplicar uma taxa de poda de 40%, o modelo comprimido obteve um aumento de acurácia de 0,28% e uma redução de 0,20 MB em seu espaço em memória ocupado, o que representa aproximadamente 30%. Para a combinação de quantização FP16 com uma taxa de poda de 40%, o aumento de acurácia foi de 0,29% e a redução em termos de requisitos de memória foi de aproximadamente 63%. É válido destacar, também, os resultados obtidos ao combinar a técnica de quantização de faixa dinâmica com taxas de poda de 40% e 60%. Para os dois casos a queda de acurácia foi de 0,55%. Em relação ao espaço em memória ocupado, para taxa de poda de 40%, o modelo comprimido foi reduzido em aproximadamente 78%, enquanto que para taxa de 40%, essa redução foi de aproximadamente 83%

Em relação aos resultados de acurácia obtidos para taxa de ocultação de 0,2 bpp apresentados na Figura 9, é possível afirmar que vários modelos comprimidos apresentaram valores iguais ou levemente superiores quando comparados com o modelo original. Esses resultados foram obtidos a partir da utilização das estratégias de poda com taxa de 60% e 70%, quantização de faixa dinâmica e combinação de quantização FP16 e taxas de poda de 60% e 70%. Além disso, outros quatro cenários apresentaram resultados com quedas de acurácia

inferiores a 1%. Destaca-se o resultado obtido ao utilizar a estratégia de quantização do tipo FP16 combinado com uma taxa de poda de 70%. O modelo comprimido apresentou um resultado de acurácia levemente superior ao do GBRAS-Net e uma redução de espaço em memória ocupado pelo modelo de aproximadamente 75% quando comparado ao modelo original.

VI. Conclusão

O presente trabalho teve como objetivo a avaliação de técnicas de compressão de modelos de aprendizado profundo aplicados à esteganálise de imagens digitais, mais especificamente com a utilização de estratégias de quantização, poda e compartilhamento de pesos. Foram avaliados os resultados em termos de acurácia e de espaço em memória ocupado pelo modelo comprimido. Além disso, foi realizada uma combinação entre os métodos de quantização e poda.

Foram avaliados os cenários da detecção de esteganografia em imagem a partir dos algoritmos MiPOD, HILL e HUGO para taxas de ocultação de 0,2 bpp e 0,4 bpp.

Foi possível obter modelos comprimidos com valores de acurácia próximos ou levemente superiores aos apresentados pelo modelo original para cada cenário envolvendo a detecção de esteganografia a partir dos algoritmos de esteganografia considerados neste trabalho.

Em relação a trabalhos futuros, é válido analisar a implementação de diferentes algoritmos de agrupamento na utilização da estratégia de compartilhamento de pesos. Dentre os algoritmos, é possível citar a utilização de versões modificadas do *k-means* [36] ou *fuzzy k-means* [37] e inteligência de enxames [38]. Além disso, vem sendo comum a utilização de *Vision Transformers* (ViT) para resolução de diversas tarefas visuais. Por esse motivo, é válido avaliar a utilização dessa técnica aplicada à esteganálise, bem como analisar a viabilidade da implementação de técnicas de compressão aplicadas a ViT.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001 e do CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico.

Referências

[1] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007.

[2] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, 1st ed. USA: Cambridge University Press, 2009.

[3] H. Zha, W. Zhang, N. Yu, and Z. Fan, "Enhancing image steganography via adversarial optimization of the stego distribution," *Signal Processing*, vol. 212, p. 109155, 2023.

[4] J. Kunhoth, N. Subramanian, S. Al-Maadeed, and A. Bouridane, "Video steganography: Recent advances and challenges," *Multimedia Tools and Applications*, vol. 82, no. 27, pp. 41 943–41 985, apr 2023.

[5] Y. Ren, D. Liu, C. Liu, Q. Xiong, J. Fu, and L. Wang, "A universal audio steganalysis scheme based on multiscale spectrograms and DeepResNet," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 1, pp. 665–679, 2023.

[6] J. Wen, X. Zhou, P. Zhong, and Y. Xue, "Convolutional neural network based text steganalysis," *IEEE Signal Processing Letters*, vol. 26, no. 3, pp. 460–464, 2019.

[7] T. Muralidharan, A. Cohen, A. Cohen, and N. Nissim, "The infinite race between steganography and steganalysis in images," *Signal Processing*, vol. 201, p. 108711, 2022.

[8] E. Zielińska, W. Mazurczyk, and K. Szczypiorski, "Trends in steganography," *Communications of the ACM*, vol. 57, no. 3, pp. 86–95, 2014.

[9] S. Alneyadi, E. Sithirasenan, and V. Muthukkumarasamy, "A survey on data leakage prevention systems," *Journal of Network and Computer Applications*, vol. 62, pp. 137–152, 2016.

[10] H. Wang and S. Wang, "Cyber warfare: steganography vs. steganalysis." *Communications of the ACM*, vol. 47, pp. 76–82, 01 2004.

[11] S. Rallabandi, A. M. Sundaram, and K. V, "Generating a multi-OS fully undetectable malware (FUD) and analyzing it afore and after steganography," in *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 2022, pp. 1962–1967.

[12] L. Almehmadi, A. Basuhail, D. Alghazzawi, and O. Rabie, "Framework for malware triggering using steganography," *Applied Sciences*, vol. 12, no. 16, 2022.

[13] K. Karampidis, E. Kavallieratou, and G. Papadourakis, "A review of image steganalysis techniques for digital forensics," *Journal of Information Security and Applications*, vol. 40, pp. 217–235, 2018.

[14] R. Ibrahim and T. S. Kuan, "Pris: Image processing tool for dealing with criminal cases using steganography technique," in *2011 Sixth International Conference on Digital Information Management*, 2011, pp. 193–198.

[15] H. D. Samuel, M. S. Kumar, R. Aishwarya, and G. Mathivanan, "Automation detection of malware and stenographical content using machine learning," in *2022 6th International Conference on Computing Methodologies and Communication (ICCMC)*, 2022, pp. 889–894.

[16] N. J. D. L. Croix, T. Ahmad, and F. Han, "Comprehensive survey on image steganalysis using deep learning," *Array*, vol. 22, p. 100353, 2024.

[17] N. J. D. L. Croix and T. Ahmad, "Fuzconvsteganalysis: Steganalysis via fuzzy logic and convolutional neural network," *SoftwareX*, vol. 26, p. 101713, 2024.

[18] S. Agarwal and K.-H. Jung, "Digital image steganalysis using entropy driven deep neural network," *Journal of Information Security and Applications*, vol. 84, p. 103799, 2024.

[19] T. Fu, L. Chen, Z. Fu, K. Yu, and Y. Wang, "CCNet: CNN model with channel attention and convolutional pooling mechanism for spatial image steganalysis," *Journal of Visual Communication and Image Representation*, vol. 88, p. 103633, 2022.

[20] D. R. I. M. Setiadi, S. Rustad, P. N. Andono, and G. F. Shidik, "Digital image steganography survey and investigation (goal, assessment, method, development, and dataset)," *Signal Processing*, vol. 206, p. 108908, 2023.

[21] R. Beale and T. Jackson, *Neural computing: An introduction*. GBR: IOP Publishing Ltd., 1990.

[22] C. C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*, 1st ed. Springer Publishing Company, Incorporated, 2018.

[23] G. Ferreira, M. H. d. N. Marinho, V. Severo, and F. Madeiro, "Optimization strategies applied to deep learning models for image steganalysis: Application of pruning, quantization and weight clustering," *Applied Sciences*, vol. 15, no. 9, 2025.

[24] S. Vadera and S. Ameen, "Methods for pruning deep neural networks," *IEEE Access*, vol. 10, pp. 63 280–63 300, 2022.

[25] Z. Wang, M. Wen, Y. Xu, Y. Zhou, J. H. Wang, and L. Zhang, "Communication compression techniques in distributed deep learning: A survey," *Journal of Systems Architecture*, vol. 142, p. 102927, 2023.

[26] G. C. Marino, A. Petrini, D. Malchiodi, and M. Frasca, "Deep neural networks compression: A comparative survey and choice recommendations," *Neurocomputing*, vol. 520, pp. 152–170, 2023.

[27] T.-S. Reinel, A.-A. H. Brayan, B.-O. M. Alejandro, M.-R. Alejandro, A.-G. Daniel, A.-G. J. Alejandro, B.-J. A. Buenaventura, O.-A. Simon, I. Gustavo, and R.-P. Raúl, "GBRAS-Net: A convolutional neural network architecture for spatial image steganalysis," *IEEE Access*, vol. 9, pp. 14 340–14 350, 2021.

[28] G. d. A. Ferreira, F. A. B. S. Ferreira, M. J. C. E. d. Azevedo, V. S. d. Lima, and F. Madeiro, "Ocultação de informação em imagens digitais: uma introdução e ferramentas de apoio ao ensino," *Caderno Pedagógico*, vol. 22, no. 4, p. e14343, fev. 2025.

[29] H. Cheng, M. Zhang, and J. Q. Shi, "A survey on deep neural network pruning: taxonomy, comparison, analysis, and recommendations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 10 558–10 578, 2024.

[30] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," in *28th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'15. Cambridge, MA, USA: MIT Press, 2015, pp. 1135–1143.

[31] G. Castellano, E. Cotardo, C. Mencar, and G. Vessio, "Density-based clustering with fully-convolutional networks for crowd flow detection from drones," *Neurocomputing*, vol. 526, pp. 169–179, 2023.

[32] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Information Sciences*, vol. 622, pp. 178–210, 2023.

[33] V. Sedighi, R. Cogranne, and J. Fridrich, "Content-adaptive steganography by minimizing statistical detectability," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 2, pp. 221–234, 2016.

[34] T. Pevný, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Information Hiding*, R. Böhme, P. W. L. Fong, and R. Safavi-Naini, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 161–177.

[35] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," *2014 IEEE International Conference on Image Processing, ICIP 2014*, pp. 4206–4210, 01 2015.

[36] R. de Maeyer, S. Sieranoja, and P. Fränti, "Balanced k-means revisited," *Applied Computing and Intelligence*, vol. 3, no. 2, pp. 145–179, 2023.

[37] E. Mata, S. Bandeira, P. De Mattos Neto, W. Lopes, and F. Madeiro, "Accelerating families of fuzzy k-means algorithms for vector quantization codebook design," *Sensors*, vol. 16, no. 11, 2016.

[38] V. Severo, F. B. S. Ferreira, R. Spencer, A. Nascimento, and F. Madeiro, "On the initialization of swarm intelligence algorithms for vector quantization codebook design," *Sensors*, vol. 24, no. 8, 2024.