STEGS2GAN - Um Modelo de Aprendizado Profundo para Esteganografia e Esteganálise em Imagens Digitais utilizando Redes Adversárias Generativas

João Pedro B. S. Souza, Breno W. B. Correia, Kleber M. Trevisani

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP)
Campus Presidente Epitácio
Rua José Ramos Júnior, 27-50, CEP 19477-170 – Presidente Epitácio – SP

jpbs921@gmail.com, breno.wille@aluno.ifsp.edu.br, kleber@ifsp.edu.br

Abstract. *This work describes the development of a deep learning model to perform steganography and steganalysis on digital images. The model architecture uses a Generative Adversarial Network (GAN), whose generator is trained to imperceptibly hide textual data in images and the discriminator to detect anomalies that may indicate text hiding in images. The goal is to create a competitive training to evaluate the balance between the visual quality of the image, the recovery rate of hidden information and resistance to detection. The methodology explores the adaptation of residual deep networks using an adversarial environment to create robust hiding patterns.*

Resumo. *Este trabalho descreve o desenvolvimento de um modelo de aprendizado profundo para realizar esteganografia e esteganálise em imagens digitais. A arquitetura do modelo utiliza uma Rede Adversária Generativa (GAN), cujo gerador é treinado para ocultar dados textuais em imagens imperceptivelmente e o discriminador para detectar anomalias que podem indicar ocultação de textos em imagens. O objetivo é criar um treinamento competitivo para avaliar o equilíbrio entre a qualidade visual da imagem, a taxa de recuperação da informação oculta e a resistência à detecção. A metodologia explora a adaptação de redes profundas residuais utilizando ambiente adversarial para a criação de padrões de ocultação robustos.*

1. Introdução

A esteganografia atua na ocultação de mensagens e outros tipos de dados em mídias de cobertura (como arquivos de imagem, áudio e vídeo), diferenciando-se da criptografia, que apenas embaralha a mensagem tornando-a ininteligível. Já a esteganálise é a ciência dedicada a detectar a presença dessas informações ocultas e, se possível, extraí-las ou neutralizá-las (Fridrich, 2009). Essas áreas de estudo têm sido amplamente utilizadas na segurança da informação e na perícia forense digital. Suas aplicações variam desde a transmissão segura de dados governamentais e militares até o monitoramento de redes para a prevenção de espionagem industrial (vazamento de dados corporativos) e a identificação de comunicações ilícitas por organizações criminosas e terroristas (Kessler, 2004; Cheddad et al., 2010).

Historicamente, métodos tradicionais e estáticos de esteganografia, como a Substituição do Bit Menos Significativo (LSB), dominaram a área devido à sua simplicidade. Em contrapartida, a esteganálise desenvolveu-se como a ciência focada em detectar e expor essas comunicações ocultas. Com o avanço da Aprendizagem Profunda, ambas as áreas, esteganografia e esteganálise, sofreram uma revolução tecnológica considerável. Assim como a esteganografia passou a utilizar redes neurais para embutir dados de forma complexa, a esteganálise evoluiu drasticamente para combater essas técnicas. Ferramentas modernas de detecção passaram a utilizar modelos

de aprendizado profundo capazes de extrair anomalias estatísticas, captar ruídos residuais de alta frequência e detectar alterações invisíveis ao olho humano com altíssima precisão, criando uma verdadeira corrida contínua entre a ocultação (esteganografia) e a detecção (esteganálise).

Desenvolvido como trabalho de conclusão de curso de graduação, o objetivo deste trabalho foi desenvolver um modelo de aprendizado profundo, denominado Stegs2GAN (*Steganography and Steganalysis Dual GAN*), que utiliza Redes Adversárias Generativas (GAN) (Goodfellow et al., 2014) para realizar esteganografia e esteganálise de forma integrada, aplicando um treinamento simultâneo e competitivo considerando as duas frentes. A justificativa para este desenvolvimento apoia-se na necessidade de criar um modelo de ocultação robusto, que não apenas esconda dados visualmente, mas que seja ativamente treinado para resistir às ferramentas modernas de esteganálise.

Com relação aos trabalhos relacionados, brevemente descritos na seção 2, este trabalho apresenta duas principais contribuições. A primeira é a adição de blocos residuais ao codificador e ao decodificador, camadas que somam a entrada do bloco à sua saída transformada, com o objetivo de preservar o fluxo do gradiente e viabilizar redes mais profundas sem degradar a estabilidade do treinamento. A segunda é a substituição de um discriminador convencional por esteganalisadores que utilizam aprendizado profundo atualmente considerados estado da arte, adaptados para operar dentro do laço adversarial, de modo que o gerador seja treinado contra o mesmo tipo de detector que enfrentaria em um cenário real de esteganálise.

O modelo de ameaça para este trabalho é estruturado de forma que o emissor transmita uma imagem por um canal monitorado e o adversário observe o tráfego, decidindo apenas se a imagem contém ou não informação oculta, sem alterá-la. O receptor precisa dispor dos pesos do decodificador treinado, que operam como segredo compartilhado. Não há chave criptográfica adicional, de modo que o objetivo de segurança perseguido é a indetectabilidade da existência da mensagem, e não a confidencialidade do seu conteúdo. Assume-se que o adversário conhece a arquitetura empregada, mas não os pesos do modelo.

2. Trabalhos Relacionados

A literatura recente apresenta avanços significativos tanto na esteganografia quanto na esteganálise baseada em aprendizado profundo.

No campo da esteganálise, You, Zhang e Zhao (2020) propuseram a Siamese CNN, uma rede neural siamesa que divide a imagem em duas partes para identificar inconsistências e características de esteganografia adaptativa. Já Peng et al. (2024) desenvolveram a RS-GAN, focada na robustez em ambientes adversariais, utilizando GANs para reconstruir amostras e calcular pontuações de anomalia. Para enfrentar a escassez de dados de treinamento (problema de *Training-Images-Shortage*), Zhang et al. (2022) apresentaram a GLSNet, que utiliza geradores (incluindo arquiteturas residuais) para criar novas imagens de treinamento para esteganálise.

Na esteganografia, o trabalho de Zhang et al. (2019) introduziu o SteganoGAN, uma arquitetura focada na alta capacidade de ocultação de dados, utilizando um componente crítico para melhorar a qualidade visual e a recuperação da mensagem. Expandindo essa abordagem, Khoma e Bashkov (2025) desenvolveram o Keras-SteganoGAN, demonstrando que a remoção de correção de erros aliada a codificadores densos mantém a alta capacidade da rede. Já Qin et al. (2020) propuseram um método de esteganografia coverless baseado em GAN, que utiliza o aprendizado residual e o mapeamento de ruídos para embutir as informações secretas sem modificar diretamente a imagem.

A presente proposta difere dos trabalhos citados ao elevar o rigor do treinamento

adversarial através da utilização de uma SRNet (Boroumand, Chen e Fridrich, 2019) adaptada ou de uma PatchGAN (Isola et al., 2017) como discriminador. Ao focar em um desses modelos durante o treinamento, a abordagem permite avaliar de forma dedicada os artefatos residuais ou a integridade local da imagem, forçando uma evolução mais direcionada e robusta na capacidade de ocultação do gerador.

3. Arquitetura Proposta

A arquitetura proposta orquestra a competição entre o gerador e o discriminador, conforme ilustrado pela Figura 1. O fluxo ocorre em quatro passos. Inicialmente, a imagem de cobertura (original) e a mensagem secreta, previamente convertida em um tensor binário, são fornecidas ao codificador, que produz a imagem stego. A imagem stego gerada deve ser visualmente idêntica à imagem de cobertura, porém com uma mensagem secreta embutida em seus dados. Em seguida, a imagem stego é entregue ao decodificador, que tenta reconstruir a mensagem secreta tendo como entrada apenas a própria imagem stego, sem acesso à imagem de cobertura original. Paralelamente, o discriminador recebe as respectivas imagens de cobertura e stego e tenta classificá-las corretamente.

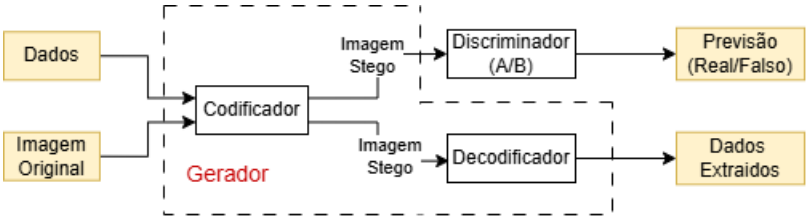

Figura 1. Arquitetura Geral

O componente gerador opera estruturado como um par de redes neurais em formato de Codificador-Decodificador, baseado na arquitetura SteganoGAN (Zhang et al., 2019), mas com a adição de blocos residuais para melhorar o fluxo de gradientes e a preservação de características. Neste âmbito, o codificador implementa o processo de esteganografia inserindo os dados secretos na imagem, conforme o detalhamento das suas camadas na Figura 2.

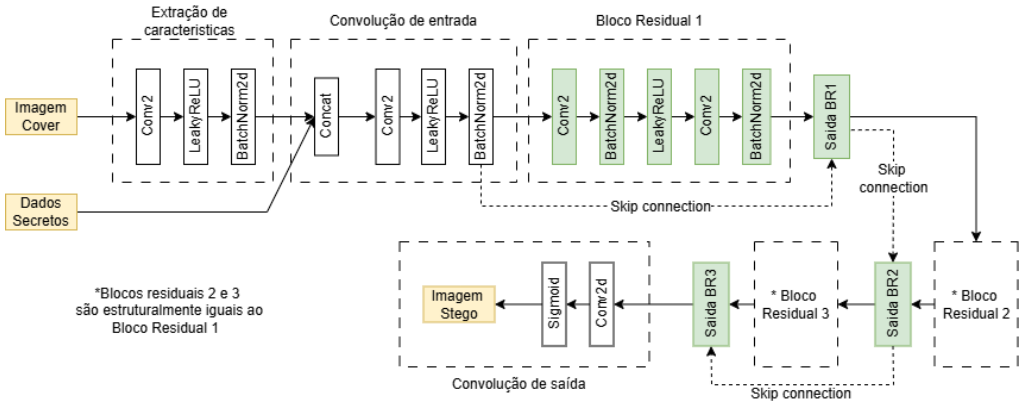

Figura 2. Arquitetura do Codificador

Na primeira etapa, o codificador recebe o tensor da imagem (D×H×W) de cobertura (*cover*) e aplica camadas convolucionais iniciais cruciais para extrair os mapas de características espaciais da imagem. Em paralelo, a mensagem secreta é processada e redimensionada para também formar um tensor no formato dos tensores de imagem (D×H×W). Na etapa intermediária, ocorre a etapa de fusão, onde o tensor da mensagem é concatenado aos mapas de características da imagem.

A partir desse ponto, o fluxo segue através de uma série de blocos residuais. A

importância desses blocos reside na capacidade de embutir os dados secretos (mensagem) profundamente nas frequências da imagem de cobertura, garantindo a ocultação sem degradar as propriedades visuais originais da imagem. Na etapa final, uma camada convolucional de saída utiliza uma função de ativação sigmoide para normalizar (utilizando valores entre 0 e 1) os *pixels* da imagem *stego* gerada.

Cada bloco residual é composto por camadas convolucionais acrescidas de uma conexão de atalho que soma a entrada à saída dessas camadas. Essa soma preserva o fluxo do gradiente em redes profundas e permite que as camadas aprendam apenas o resíduo, ou seja, a perturbação a ser somada à imagem, o que é conveniente quando a modificação desejada seja de baixa amplitude.

O decodificador, ilustrado pela Figura 3, realiza a extração dos dados secretos embutidos na imagem *stego* (recebida como entrada única), que é um processo estruturado para isolar o ruído esteganográfico e recuperar a informação. Essa etapa inicial do decodificador atua como um filtro, utilizando camadas convolucionais dedicadas a separar os padrões sutis introduzidos pela ocultação da mensagem na imagem original.

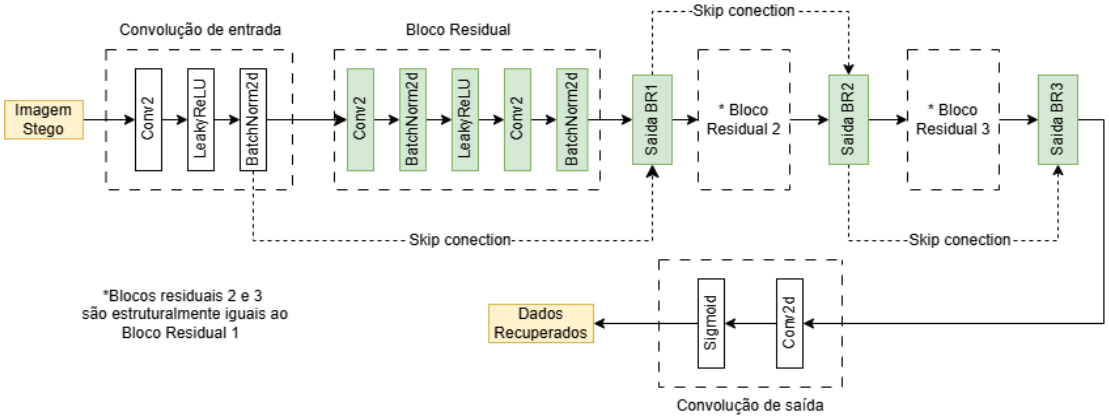

Figura 3. Arquitetura do Decodificador

Nas etapas intermediárias, o decodificador emprega blocos residuais para refinar as características extraídas, que são fundamentais para reconstruir a representação da mensagem com precisão, lidando com o ruído sem perder informações. A etapa final consiste em transformar a representação da mensagem em um tensor de saída que é posteriormente submetido a uma ativação sigmoide. Cada parte desta estrutura foi projetada para garantir que a extração da mensagem original (dados secretos) seja robusta, considerando as complexidades visuais inseridas na imagem *stego*.

Para realizar o papel adversarial, este trabalho foi implementado de forma modular, permitindo a seleção entre duas arquiteturas distintas de discriminador durante a fase de treinamento. A justificativa para esta abordagem baseia-se na necessidade de forçar a evolução do gerador sob duas perspectivas de segurança rigorosas e complementares: a qualidade visual da imagem stego e a resistência à esteganálise.

Ao optar pelo DiscriminadorPatch, baseado na arquitetura PatchGAN (Isola et al., 2017), o treinamento se concentra na integridade visual. Esta arquitetura é eficaz em capturar anomalias locais ao classificar a imagem em pequenos blocos (*patches*) de N×N, forçando o gerador a esconder a mensagem sem gerar artefatos visíveis de alta frequência. A estrutura deste discriminador, com a remoção da ativação final para estabilidade com perdas de entropia cruzada, é ilustrada pela Figura 4. Em outros termos, ao invés de emitir um único veredito para a imagem inteira, o PatchGAN emite um veredito por região, o que o torna sensível a artefatos localizados, porém pouco sensível a alterações estatísticas distribuídas por toda a imagem.

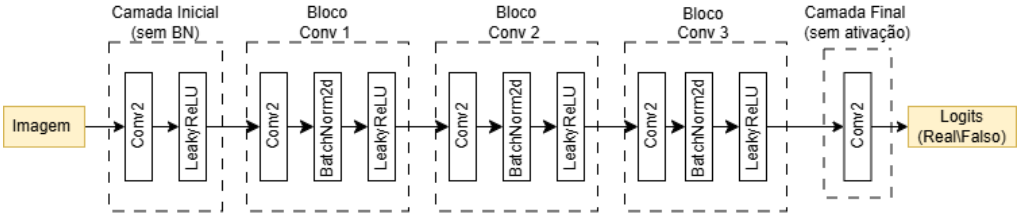

Figura 4. Arquitetura do DiscriminadorPatch

Por outro lado, ao selecionar a SRNet adaptada, o foco do treinamento passa a ser a segurança estatística. A SRNet foi concebida especificamente para capturar micro-padrões de ruído esteganográfico imperceptíveis a olho nu. Para se adequar ao fluxo da GAN, foi necessário realizar modificações significativas na SRNet: a camada de entrada foi configurada para três canais (RGB), a estrutura de agrupamento médio global foi modificada, a normalização final foi removida para preservar as estatísticas de ruído sutil, e foram feitos ajustes de *pooling* e *stride* para suportar o tamanho de entrada fixo de 360 × 360 pixels. O ruído residual corresponde ao que resta da imagem após a supressão das componentes de baixa frequência. É nele que o processo de ocultação introduz suas alterações.

4. Metodologia

A pesquisa caracteriza-se como experimental e quantitativa. A definição da arquitetura e dos hiperparâmetros operacionais (pesos das funções de perda, taxa de aprendizagem e número de épocas) foi feita de forma empírica, a partir de testes preliminares que compararam diferentes topologias e da observação da estabilidade do treinamento. Esta seção descreve detalhes de treinamento, conjuntos de dados utilizados, detalhes de pré-processamento, estratégia de divisão das amostras, definição dos hiperparâmetros, descrição do ambiente computacional utilizado e o protocolo de avaliação adotado.

Com as arquiteturas de geração e detecção devidamente estruturadas, o modelo é submetido ao processo de aprendizado. O treinamento segue um protocolo adversarial onde a cada iteração de lote (*batch*) a otimização do gerador e do discriminador selecionado são realizadas sequencialmente. A otimização é guiada por funções de perda específicas. A perda do discriminador é calculada pela média aritmética das perdas individuais após classificar imagens reais e stego, conforme a fórmula a seguir:

$$L_D = \frac{L_{Stego} + L_{Original}}{2}$$

Já a perda do gerador é calculada utilizando a função híbrida a seguir:

$$L_G = L_{dados} + \beta L_{Img} + \gamma L_{Adv}$$

onde:

L_{dados} refere-se ao grau de integridade dos dados

$L_{Img} = 1 - SSIM$ *representa o grau de imperceptibilidade*

L_{Adv} quantifica a segurança contra esteganálise

$\beta = 0{,}75 \quad e \quad \gamma = 0{,}1$ são os pesos das perdas de imagem e adversarial,

Para a condução dos experimentos, foi utilizado o conjunto de dados DIV2K (Agustsson e Timofte, 2017), de alta resolução, com 900 imagens, e uma amostragem do conjunto COCO (Lin et al., 2014), com 5 mil imagens de um total de 120 mil, reconhecido pela variabilidade de texturas. Todas as amostras passaram por recorte central de 360×360 pixels e os dois conjuntos de dados foram particionados em 70%

para treinamento, 15% para validação e 15% para teste. O trabalho foi desenvolvido em Python 3, adotando o PyTorch, integrado ao Torchvision e pytorch-MSSSIM.

A capacidade de carga foi fixada em 800 bits por imagem, o que corresponde a aproximadamente 0,0062 bits por pixel para as imagens de 360 × 360 utilizadas. As mensagens foram geradas a partir de um léxico da língua portuguesa, convertidas em tensores binários e remodeladas para as dimensões da imagem. Como as frases raramente ocupam toda a capacidade disponível, uma máscara binária identifica os bits de informação real, de modo que o preenchimento receba peso reduzido na perda e seja desconsiderado nas métricas de recuperação.

Quatro configurações foram treinadas e testadas, combinando os dois conjuntos de dados com os dois tipos de discriminadores. Os tempos de treinamento foram de aproximadamente 2,5 e 4 horas no DIV2K e de 10 e 20 horas no COCO, respectivamente com o DiscriminadorPatch e com a SRNet.

O treinamento foi executado em nuvem com uma GPU NVIDIA Tesla T4 (16 GB de VRAM), utilizando lotes de 8 imagens. Após testes preliminares, o processo foi padronizado em 100 épocas, com taxa de aprendizagem de 1×10^{-3} para ambas as redes.

Ao final de cada época, o modelo foi avaliado sobre o conjunto de validação e os pesos foram salvos sempre que a perda total de validação atingia um novo mínimo. Os resultados reportados na seção 5 referem-se, portanto, ao modelo de menor perda de validação, e não ao modelo da última época, evitando que degradações ocorridas ao final do treinamento sejam incorporadas à avaliação. O código-fonte do trabalho pode ser obtido em https://github.com/jpbss/STEGS2GAN.

5. Resultados

Os experimentos visam validar a hipótese de que um treinamento adversarial contra um discriminador robusto força o gerador a produzir esteganografia de alta fidelidade. A seguir, os resultados são discutidos sob a ótica da qualidade visual, segurança e estabilidade do treinamento.

5.1. Análise da Qualidade de Ocultação e Recuperação

A eficácia do Gerador foi medida pela sua capacidade de enganar o discriminador mantendo a integridade da imagem e da mensagem (dados ocultos). Para melhor compreensão do desempenho alcançado, os resultados foram consolidados na Tabela 1 considerando quatro métricas principais. A qualidade visual da imagem stego foi avaliada pelo PSNR, que mede a fidelidade do sinal em decibéis, onde valores acima de 30 dB indicam uma distorção física aceitável, e pelo SSIM (Wang et al., 2004), que quantifica a similaridade estrutural alinhada à percepção humana em uma escala de 0 a 1 (sendo 1 a identidade perfeita).

Já a eficácia na extração da informação oculta foi mensurada pela Acurácia de Bits, que indica o percentual médio de bits recuperados corretamente, e pela rigorosa Acurácia de Mensagem, que aponta o percentual de mensagens recuperadas sem absolutamente nenhum erro, onde a falha de um único bit invalida a recuperação total da mensagem.

Cabe ressaltar que as métricas PSNR e SSIM permitem avaliar respectivamente a fidelidade do sinal e a similaridade estrutural em relação à imagem de cobertura, isto é, a imperceptibilidade para o sistema visual humano. Essas métricas não constituem, isoladamente, evidência de segurança esteganográfica: uma imagem *stego* pode ser visualmente indistinguível da original e ainda assim apresentar assinaturas estatísticas detectáveis. A avaliação de segurança é conduzida separadamente, por meio do desempenho dos esteganalisadores, na subseção seguinte.

A interpretação dos dados apresentados na Tabela 1 sugere que a rigidez do

adversário tende a elevar a qualidade do gerador. A configuração que utilizou a SRNet (um discriminador mais profundo e complexo) resultou em métricas de qualidade visual superiores (SSIM de 0,9831) quando comparada ao DiscriminadorPatch (SSIM de 0,9720). Isso sugere que, ao enfrentar um crítico com melhor capacidade de detectar micro-ruídos, o Gerador foi forçado a aprender estratégias de ocultação mais sofisticadas e menos perceptíveis para conseguir convergir.

Tabela 1. Desempenho do Gerador (Qualidade Visual e Recuperação)

Configuração	Amostras de imagem	PSNR (dB)	SSIM	Acurácia de Bits (%)	Acurácia de Mensagem (%)
DIV2K + SRNet	135	31,15 ± 3,73	0,9831 ± 0,0122	99,98 ± 0,11	97,04
DIV2K + DiscriminadorPatch	135	30,57 ± 2,69	0,9720 ± 0,0245	99,99 ± 0,05	95,56
COCO + SRNet	750	29,32 ± 2,59	0,9801 ± 0,0161	99,99 ± 0,05	93,47
COCO + DiscriminadorPatch	750	26,66 ± 2,10	0,9580 ± 0,0241	99,85 ± 0,47	64,93

Além disso, nota-se a influência crítica da homogeneidade do conjunto de dados. O DIV2K, composto por imagens de alta resolução e qualidade técnica consistente, facilitou o aprendizado de padrões estáveis de ocultação, resultando na maior taxa de recuperação de mensagens (97,04%). Em contraste, a alta variabilidade do conjunto COCO introduziu ruídos que dificultaram a convergência, degradando severamente a recuperação de mensagens na configuração com DiscriminadorPatch (64,93%).

Um exemplo de imperceptibilidade visual alcançada na melhor configuração no âmbito deste trabalho (Div2k + SRNet) é ilustrado pela Figura 5, onde a amplificação em 10 vezes da diferença entre a imagem original e a imagem stego demonstra que as alterações se concentram em regiões de textura, explorando a deficiência do sistema visual humano.

Figura 5. Comparação visual DIV2K + SRNet

Embora a configuração COCO + SRNet não tenha sido a de melhor desempenho geral no âmbito deste trabalho, ela ainda possibilitou a geração de imagens stego difíceis de serem detectadas pelo sistema visual humano, como pode ser observado no exemplo apresentado na Figura 6.

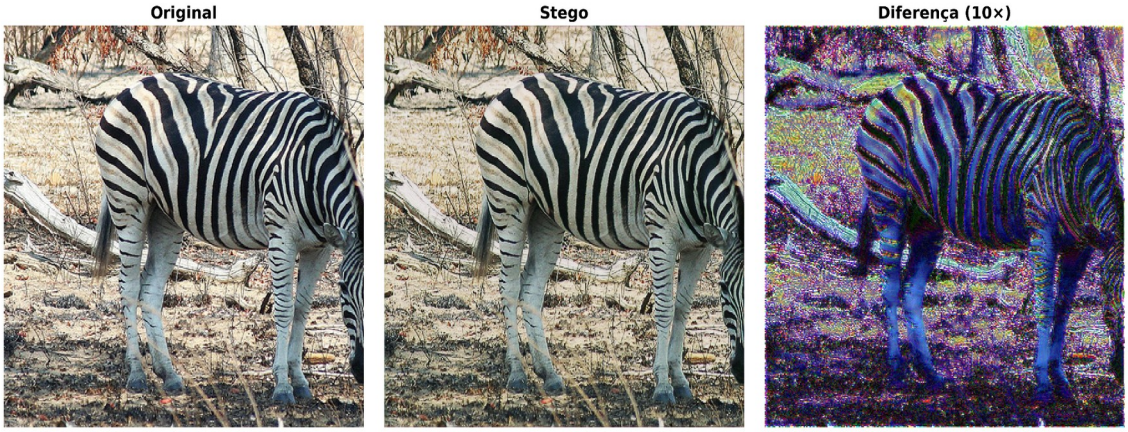

Figura 6. Comparação visual COCO + SRNet

5.2. Análise da Robustez Adversarial

A segurança da esteganografia é confirmada quando o discriminador falha em distinguir imagens reais de imagens *stego*, resultando em uma classificação aleatória. A Tabela 2 detalha este comportamento. A AUC (*Area Under Curve*) corresponde à probabilidade do detector atribuir a uma imagem stego escolhida ao acaso um *score* superior ao de uma imagem de cobertura também escolhida ao acaso. Diferentemente da acurácia, ela independe do limiar de decisão adotado, e o valor 0,5 corresponde ao desempenho de um classificador aleatório.

Tabela 2. Desempenho dos Discriminadores no Conjunto de Teste

Configuração	Acurácia (%)	Precisão (%)	Recall (%)	F1 Score (%)	AUC
DIV2K + DiscriminadorPatch	49,26	49,59	89,63	63,85	0,4758
DIV2K + SRNet	50,00	50,00	100,00	66,67	0,4971
COCO + DiscriminadorPatch	50,00	50,00	100,00	66,67	0,3055
COCO + SRNet	42,53	31,46	12,67	18,06	0,3838

A acurácia de 50% não constitui, isoladamente, evidência de segurança: um classificador que atribui a mesma classe a todas as amostras alcança esse valor sem ter aprendido qualquer critério de separação. É necessário, portanto, distinguir duas explicações possíveis para o resultado: que o gerador tenha produzido imagens stego estatisticamente indistinguíveis das de cobertura, ou que o discriminador simplesmente não tenha convergido.

Para a configuração DIV2K + SRNet, três evidências independentes sustentam a primeira explicação. A primeira é a AUC de 0,4971, que por ser calculada sobre os *scores* contínuos do discriminador e independer do limiar de decisão, ela indica que as distribuições de *scores* atribuídos às imagens de cobertura e às imagens stego se sobrepõem, ou seja, que não existe limiar capaz de separá-las. A segunda é o comportamento da perda adversarial ao longo do treinamento, apresentado na Figura 7(a), em que as perdas do gerador e do discriminador oscilam de forma estável em torno de $\ln 2 \approx 0{,}693$, valor que corresponde ao ponto de equilíbrio teórico de uma GAN, sem que nenhuma das redes domine a disputa. A terceira é o contraste com a configuração COCO + SRNet, discutida a seguir, cujo comportamento de treinamento é qualitativamente distinto sob os mesmos indicadores, o que demonstra que os dois

regimes são distinguíveis.

Observa-se, ainda, que a acurácia de exatos 50,00% dessa configuração decorre de um discriminador que passou a classificar todas as amostras do conjunto de teste como *stego*, sem estabelecer um limite de separação válido. Para o objetivo perseguido esse comportamento é favorável, pois indica que o detector foi neutralizado; a evidência que sustenta essa interpretação, contudo, é a AUC, e não a acurácia.

Valores de AUC bem abaixo de 0,5 também não significam indetectabilidade, pois a métrica é simétrica. Ficar abaixo de 0,5 traz a mesma informação que ficar acima, o detector apenas aprendeu a separação com os rótulos invertidos, bastando inverter a regra de decisão para corrigi-lo. Os testes no COCO mostram isso: as AUCs de 0,3055 (DiscriminadorPatch) e 0,3838 (SRNet) equivalem a 0,6945 e 0,6162 após a inversão. Nesses casos, as imagens *stego* e de cobertura continuam estatisticamente separáveis, mesmo que a acurácia isolada de 50,00% do primeiro sugira o oposto. Já no DIV2K, as distribuições de *scores* realmente se sobrepõem

A configuração COCO + SRNet, por outro lado, sofreu colapso de modo. O colapso de modo ocorre quando o treinamento adversarial deixa de progredir de forma equilibrada e converge para uma solução degenerada, na qual uma das redes deixa de fornecer gradientes úteis à outra e a diversidade dos dados deixa de ser aprendida. Nesta configuração, o fenômeno foi identificado por três indicadores concomitantes, e não pela acurácia final: a perda total de validação atingiu seu mínimo já na 7ª época (0,0873) e passou a divergir a partir da 10ª; a perda adversarial abandonou o patamar de equilíbrio de ln 2 e passou a oscilar em valores muito superiores, com amplitude crescente ao longo das épocas (Figura 7(b)), e a qualidade visual das imagens geradas passou a se degradar. A AUC de 0,3838 e o recall de 12,67% observados ao final são consequência desse processo, e não o critério que o caracteriza. Neste caso, o gerador "venceu" não por produzir uma ocultação melhor, mas pela incapacidade do discriminador de aprender características generalizáveis.

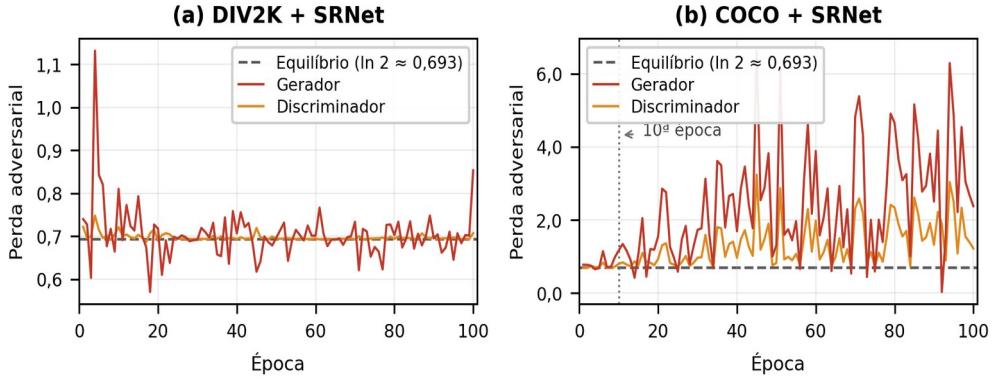

Figura 7. Perda adversarial ao longo das épocas: (a) DIV2K + SRNet, com gerador e discriminador oscilando em torno do equilíbrio teórico ln 2 ≈ 0,693; (b) COCO + SRNet, com divergência progressiva a partir da 10ª época.

6. Conclusões

O presente trabalho atingiu seu objetivo geral de desenvolver um modelo de aprendizado profundo baseado em GAN capaz de realizar esteganografia e esteganálise de forma integrada e simultânea. A validação deste objetivo é evidenciada pela confirmação empírica do equilíbrio adversarial na configuração otimizada (DIV2K + SRNet): a AUC de 0,4971, medida independente do limiar de decisão e a estabilização da perda adversarial em torno de ln 2 indicam que o discriminador não dispõe de critério estatístico capaz de separar imagens de cobertura de imagens stego, demonstrando que o treinamento competitivo forçou o gerador a produzir ocultações sem assinaturas exploráveis por esse discriminador.

Apesar do êxito no objetivo central, foram identificados limites de aplicabilidade. A homogeneidade visual dos dados se mostrou crítica para a estabilidade do treinamento. Enquanto o conjunto DIV2K favoreceu a convergência, a alta variabilidade do conjunto COCO induziu ao fenômeno de colapso de modo em arquiteturas profundas. Adicionalmente, a capacidade de carga estática restringiu a eficiência do modelo, impedindo o aproveitamento ideal de regiões texturizadas.

Para trabalhos futuros, sugere-se a adoção da arquitetura WGAN desenvolvida por Arjovsky, Chintala, Bottou (2017) para mitigar a instabilidade de gradientes em domínios complexos, além da incorporação de mecanismos de atenção visual para distribuir a carga de dados de forma adaptativa à complexidade textural da imagem. Recomenda-se também investigar o impacto da expansão do conjunto de dados no processo de treinamento, avaliando por exemplo, se utilização de imagens com resoluções maiores ou se o aumento expressivo de amostras do conjunto COCO pode fornecer a densidade informacional necessária para aprimorar a generalização do modelo e, consequentemente, mitigar o colapso de modo diante de texturas altamente variáveis.

A avaliação de segurança aqui apresentada restringe-se ao discriminador que participou do laço adversarial, o que estabelece indetectabilidade em relação a esse adversário específico, mas não a esteganalisadores arbitrários. Da mesma forma, não foi conduzido um estudo de ablação que isolasse a contribuição dos blocos residuais em relação à arquitetura que serviu de base, nem foram incluídos baselines externos, como o SteganoGAN original ou métodos clássicos de substituição do bit menos significativo. Essas três avaliações permanecem como continuidade a ser explorada em trabalhos futuros.

7. Agradecimentos

Os autores agradecem ao Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) – Campus Presidente Epitácio por todo o apoio à realização e à publicação deste trabalho.

Referências

Agustsson, E., Timofte, R. (2017) “NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study”. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, p. 1122-1131.

Arjovsky, M., Chintala, S., Bottou, L. (2017) “Wasserstein GAN”. Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, Austrália, p. 214-223. Disponível em: https://arxiv.org/abs/1701.07875.

Boroumand, M., Chen, M., Fridrich, J. (2019) “Deep Residual Network for Steganalysis of Digital Images”, IEEE Transactions on Information Forensics and Security, v. 14, n. 5, p. 1181-1193.

Cheddad, A., Condell, J., Curran, K., Mc Kevitt, P. (2010) “Digital image steganography: Survey and analysis of current methods”. Signal Processing, v. 90, n. 3, p. 727-752.

Fridrich, J. (2009) Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge: Cambridge University Press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014) “Generative Adversarial Nets”. Advances in Neural Information Processing Systems (NIPS), v. 27, p. 2672-2680.

Isola, P., Zhu, J., Zhou, T., Efros, A. (2017) “Image-to-Image Translation with

Conditional Adversarial Networks". 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, p. 5967-5976.

Kessler, G. (2004) "An Overview of Steganography for the Computer Forensics Examiner". Forensic Science Communications, v. 6, n. 3, p. 1-27.

Khoma, D., Bashkov, Y. (2025) "Enhanced Image Steganography with Keras-SteganoGAN: A TensorFlow-Based GAN", Scientific Papers of Donetsk National Technical University. Series: Computer Engineering and Automation, v. 3, n. 5, p. 48-59.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L. (2014) "Microsoft COCO: Common Objects in Context". European Conference on Computer Vision (ECCV), Zurique, Suíça, p. 740-755.

Peng, Y., Yu, O., Fu, G., Zhang, W., Duan, C. (2024) "Improving the robustness of steganalysis in the adversarial environment with generative adversarial network", Journal of Information Security and Applications, v. 82, 103743.

Qin, J., Wang, J., Tan, Y., Huang, H., Xiang, X., He, Z. (2020) "Coverless Image Steganography Based on Generative Adversarial Network". Mathematics, v. 8, n. 9, 1394.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E. (2004) "Image Quality Assessment: From Error Visibility to Structural Similarity", IEEE Transactions on Image Processing, v. 13, n. 4, p. 600-612.

You, W., Zhang, H., Zhao, X. (2020) "A Siamese CNN for Image Steganalysis", IEEE Transactions on Information Forensics and Security, v. 16, p. 291-306.

Zhang, H., Song, Z., Xing, Q., Feng, B., Lin, X. (2022) "A Generative Learning Steganalysis Network against the Problem of Training-Images-Shortage". Electronics, v. 11, n. 20, 3331.

Zhang, K., Cuesta-Infante, A., Xu, L., Veeramachaneni, K. (2019) "SteganoGAN: High Capacity Image Steganography with GANs". arXiv preprint, arXiv:1901.03892.