Universidade do Minho
Escola de Engenharia

Universidade do Minho
Escola de Engenharia
Mestrado em Engenharia Informática

Unidade Curricular de Visão por Computador e Processamento de Imagem

Ano Letivo de 2024/2025

Trabalho Individual - VCPI

Martim Redondo
57889

Junho, 2025

Índice

1. Introdução e Configuração Inicial

Este trabalho implementa um *CycleGAN* para transformação **Real↔Cartoon**, com foco na exploração sistemática de *loss functions* e implementação de técnicas *state-of-the-art* para melhorar estabilidade e qualidade.

Objetivo principal: Superar limitações do *CycleGAN default* através de otimizações técnicas modernas, validando empiricamente cada melhoria implementada.

1.1. Datasets

Os **Datasets** usados foram:

- CelebA-HQ: https://www.kaggle.com/datasets/denislukovnikov/celebahq256-images-only?resource=download-directory
- Google Cartoon Set: https://www.kaggle.com/datasets/brendanartley/cartoon-faces-googles-cartoon-set

O balanceamento foi implementado garantindo:
- Paridade quantitativa entre domínios (30k imagens cada);
- Amostragem aleatória estratificada;
- Divisão consistente treino/teste (80%/20%).

1.2. Arquitetura Base

- Generator: *ResNet-based* com 6 *residual blocks*, *reflection padding*;
- Discriminator: *PatchGAN* com 70×70 *receptive field*;
- Training:*Adam optimizer*, *batch size* 1, 50 épocas planejadas.

2. Implementação Inicial e Descoberta do Primeiro Problema

2.1. Baseline com BCE Loss

A implementação inicial utilizou *BCELoss* como função adversarial padrão. Durante as primeiras épocas de treinamento, foram identificados dois problemas críticos que comprometiam completamente a funcionalidade do modelo.

2.2. Problema 1: Morte dos Discriminadores

Indício: Discriminadores convergiam para valores próximos de zero (D_A: 0.002, D_B: 0.001), sendo incapazes de distinguir entre imagens reais e falsas. O que trazia alguns problemas, como:

- **Mode collapse total**: Geradores produziam apenas ruído colorido sem estruturas reconhecíveis;
- **Generator loss elevado**: Valores entre 5.4-9.6, o que aponta que havia dificuldade para aprender;
- **Cycle loss "artificialmente" baixo**: Valores entre 0.04-0.08, sugerindo que o modelo estava a "contornar" o problema (A→B→A consistente, mas com imagens inválidas)

Motivos:

- **Training ratio desequilibrado**: 5 D-steps por 1 G-step, o que levava à criação de discriminadores muito poderosos;
- **Learning rates inadequados**: LR_G = 0.00005, enquanto LR_D = 0.0002.
- **Hyperparameters extremos**: LAMBDA_CYCLE = 200.0 (20x superior ao paper original);
- **Arquitetura discriminador over-engineered**: Spectral normalization + AdaptiveAvgPool criava discriminadores "perfeitos".

Soluções:

As soluções, depois de descobrir os problemas, foram de fácil implementação.

- **Múltiplos training steps**: 2 steps para cada discriminador por batch para "ressuscitar"
- **Rebalanceamento de ratios**: Transição para 1:1 G:D ratio;
- **Learning rate adjustment**: Equalização para LR=0.0002 para todos os componentes.

Resultado prático: Discriminadores recuperaram funcionalidade (valores 0.1-0.9), contudo surgiu outro problema envolvendo-os:

2.3. Problema 2: Discriminador B "Super-Expert"

Indício: D_B (discriminador de cartoons) atingia, rapidamente, accuracy superior a 95%, enquanto D_A (discriminador de faces reais) ficava equilibrado em 85%.

Motivo: O dataset de cartoons apresenta maior homogeneidade visual comparado ao dataset de faces reais, tornando a discriminação de "fake cartoons" muito mais simples que a discriminação de "fake reals".

Consequência: Esta assimetria resultava em *generator collapse* na direção A→B (Real→Cartoon), com o gerador incapaz de competir contra o discriminador dominante.

Soluções: Para tentar resolver este problema de dominância do D_B, foram implementadas e testadas diversas técnicas de balanceamento, como:

- **Noise Injection Variável**: Adição de ruído aleatório (foram tentados diferentes níveis de ruído → 0.05-0.15) às imagens de entrada do D_B para reduzir *over-confidence*;
- **Skip Training Assimétrico**: D_B treina apenas 1 vez a cada 4 *batches* para limitar aprendizagem excessiva;
- **Target Smoothing Específico**: Labels 0.75/0.25 em vez de 1.0/0.0 para *targets* menos confiantes;
- **Gradient Clipping**: Limitação de gradientes para prevenir atualizações excessivas.

Resultado prático: Combinação destas técnicas conseguiu balancear D_B (real: 0.47-0.88, fake: 0.17-0.30), quando comparado, com o estado anterior (real: 0.95+, fake: 0.05).

3. Evolução para LSGAN

3.1. Motivação para Mudança

Durante os vários testes, confirmou-se que o *BCELoss* levava à saturação de gradientes quando os discriminadores atingiam alta *confidence*, que consequentemente, levava a gradientes informativos insuficientes para o treino dos geradores.

3.2. Implementação LSGAN

Intuito: LSGAN utiliza *MSE loss* em vez de *BCE*, fornecendo gradientes mais informativos mesmo quando discriminadores são confiantes.

Resultado prático vs expectativa: A migração para LSGAN efetivamente resolveu a saturação de gradientes, mas revelou um novo problema - os pesos de *cycle consistency* e *identity loss* tornaram-se insuficientes devido aos gradientes mais suaves do LSGAN.

Ajuste necessário:
- Aumento de LAMBDA_CYCLE de 10.0 para 50.0;
- LAMBDA_IDENTITY de 0.5 para 25.0.

Assim consegue-se manter o equilíbrio entre *losses adversariais* e de reconstrução.

4. Otimizações Finais State-of-the-Art

4.1. Spectral Normalization

Intuito: Controlar *Lipschitz constant* dos discriminadores para prevenir gradientes instáveis durante treinamento LSGAN.

Resultado prático: Estabilização significativa do treinamento, com redução visível de *artifacts* e *mode collapse*.

4.2. TTUR (Two Time-Scale Update Rule)

Intuito: Compensar o facto de LSGAN tornar discriminadores potencialmente dominantes através de learning rates assimétricos.

Implementação: Novas variáveis → LR_G = 1e-4, LR_D = 4e-4, BETA1 = 0.0, BETA2 = 0.9

Resultado prático: Discriminadores mantiveram-se informativos sem dominar geradores, validando a *approach* teórica.

4.3. Eliminação de Checkerboard Artifacts

Intuito: Resolver *artifacts* visuais causados por *ConvTranspose2d* que eram amplificados pela sensibilidade do LSGAN.

Implementação: Substituição do *ConvTranspose2d* por *Upsample + Conv2d*

Resultado prático: Eliminação completa de padrões *checkerboard*, especialmente visível na direção B→A.

5. Convergência e Resultados Finais

5.1. Progressão Durante Treinamento

O modelo convergiu na época 25 com os seguintes indicadores:

- **Generator Loss**: 4.36 → 3.22 (melhoria 26%);
- **Cycle Loss**: 0.136 → 0.089 (excelente preservação de identidade);
- **Discriminadores**: Balanceados (pred_real 0.85, pred_fake 0.15);
- **Plateau identificado**: Épocas 21-25 sem melhoria significativa.

5.2. Qualidade Visual Observada

- **A→B**: "máscaras coloridas", não "cartoons naturais";
- **B→A**: "faces reconhecíveis" mas "algumas distorções";
- **Cycles**: Confirmado pelas métricas.

6. Avaliação Quantitativa

6.1. Métricas Implementadas

- **Cycle Consistency**: L1 loss entre imagem original e cycle reconstruction;
- **FID (Fréchet Inception Distance)**: Qualidade perceptual usando **InceptionV3** (1000 amostras);
- **LPIPS (Learned Perceptual Similarity)**: Similaridade perceptual usando AlexNet;

- **IS (Inception Score)**: Diversidade e qualidade das amostras geradas.

6.2. Resultados Quantitativos

Métrica	Val. Obtido	Baseline Literatura	Status
Cycle Consistency	0.031	CycleGAN: 0.15	Excelente
FID A→B	254.65	AttentionGAN: 200	Moderado
FID B→A	262.94	UNIT: 250	Moderado
LPIPS A→B	0.535	Típico: 0.1-0.3	Alto
LPIPS B→A	0.751	Típico: 0.1-0.3	Alto
IS Fake A	3.13±0.16	Adequado: >2.0	Bom
IS Fake B	2.33±0.17	Adequado: >2.0	Bom

Tabela 1: Principais métricas avaliadas com:
A→Real
B→Fake/Cartoon

7. Discussão e Análise de Resultados

7.1. Sucessos Validados

O resultado mais significativo é o Cycle Consistency de 0.031, 5x superior ao CycleGAN original (0.15), validando empiricamente que todas as otimizações implementadas foram eficazes na preservação de identidade.

7.2. Interpretação das Limitações

Os valores elevados de FID (255) e LPIPS (0.6) não representam falhas do modelo, mas sim a natureza extrema da transformação Real↔Cartoon.

Estas métricas capturam a magnitude da mudança perceptual necessária, enquanto o cycle consistency baixo confirma que a identidade é preservada durante o processo.

7.3. Assimetria B→A vs A→B

A superioridade da direção B→A (Cartoon→Real) sobre A→B (Real→Cartoon) é consistente com a literatura, representando a diferença entre uma “completion task” (adicionar detalhes realistas) versus uma “abstraction task” (remover detalhes seletivamente).

7.4. Técnicas Testadas mas Descartadas

- **Mixed Precision AMP**: Causou instabilidade numérica;

- **Learning rate scheduling agressivo**: Prejudicou convergência;
- **Noise injection constante**: Menos eficaz que noise variável.

8. Configuração final validada:

```
------------------------------------------------------------
    LAMBDA_CYCLE = 50.0, LAMBDA_IDENTITY = 25.0
    LR_G = 1e-4, LR_D = 4e-4, BETA1 = 0.0, BETA2 = 0.9
    Spectral Normalization + LSGAN + TTUR
------------------------------------------------------------
```