

Classificação de Deepfakes no Domínio da Frequência com Arquitetura Adaptada da F3-Net

Marinaldo Lima Pereira, Ronaldo F. Zampolo e Ana Carolina Q. Siravenha Müller

Resumo— Este artigo apresenta a reprodução do modelo F³-Net para classificação binária de *deepfakes* em vídeos, utilizando filtros adaptativos via Transformada Cosseno Discreta para evidenciar artefatos de manipulação. A metodologia incorpora otimizações de memória e pré-processamento, além de avaliar a redução do conjunto de entrada em aproximadamente 92%, de 270 para apenas 20 *frames* e da implementação customizada do módulo de atenção cruzada com apenas um ramo de interconexão. A avaliação nos subconjuntos C23 e C40 do FaceForensics++, no DFDC e em validação cruzada demonstra desempenho competitivo, tornando o método viável em cenários com restrições de tempo ou recursos computacionais.

Palavras-Chave— Detecção de falsificação de rostos, Análise Espectral, Compressão de Vídeo, Restrição de Hardware.

Abstract— This paper presents a reproduction of the F³-Net model for binary deepfake video classification, using adaptive filters based on the Discrete Cosine Transform (DCT) to highlight manipulation artifacts. The proposed methodology incorporates memory and preprocessing optimizations, and evaluates the impact of reducing the input set by approximately 92%—from 270 to only 20 frames—as well as a customized implementation of the cross-attention module using a single interaction branch. The evaluation on the C23 and C40 subsets of FaceForensics++, the DFDC dataset, and in cross-dataset validation demonstrates competitive performance, making the proposed approach suitable for scenarios with limited computational resources or strict time constraints.

Keywords— Face Forgery Detection, Spectral Analysis, Video Compression, Hardware Restriction.

I. INTRODUÇÃO

Os recentes avanços em Inteligência Artificial têm impulsionado a proliferação de mídias sintéticas e manipuladas, abalando a confiabilidade dos registros visuais digitais. Nesse cenário, o fenômeno dos *deepfakes* representa o estado da arte em manipulação visual, utilizando arquiteturas de *deep learning*, como Redes Adversárias Generativas (GANs) e *Autoencoders*, para alcançar um realismo inédito, muitas vezes além do alcance da percepção humana. Embora essas tecnologias possuam diversas aplicações, o seu uso para desinformação em redes sociais representa um risco agudo à autenticidade digital [1].

Apesar do surgimento de diversas técnicas de detecção, a aplicação prática desses métodos enfrenta um entrave técnico crucial: a compressão de vídeo. Plataformas como *WhatsApp* e *Facebook* utilizam algoritmos de compressão (H.264 e JPEG,

por exemplo) que descartam intencionalmente detalhes de alta frequência para otimizar o armazenamento. Modelos de detecção restritos ao domínio espacial, como a *MesoNet* [2] e descritores de textura tradicionais, mostram-se eficazes em vídeos de alta qualidade, mas degradam-se drasticamente em presença de compressão [3].

Além da vulnerabilidade à compressão, a complexidade computacional dos detectores de manipulação baseados em redes neurais profundas constitui um fator limitante. Arquiteturas do estado da arte demandam elevado custo em termos de operações de ponto flutuante e alocação de memória de vídeo (VRAM) [4]. Para sistemas operando sob restrição de *hardware*, o processamento de modelos densos torna-se inviável, exigindo estratégias de otimização que equilibrem robustez e eficiência computacional.

Neste contexto, este trabalho adapta a arquitetura F³-Net, proposta por Qian *et al.* [5], que utiliza o processamento de sinais no domínio da frequência (por meio da Transformada Discreta de Cosseno (DCT)) para assegurar robustez contra a compressão. As adaptações promovidas tratam da implementação do protocolo de pré-processamento de maneira *offline*. Além disso, das duas unidades de *Transformers* previstas para trocar informações entre as redes convolucionais (CNNs) apenas uma foi implementada. Nesse cenário de adaptação, avalia-se também a redução da quantidade de *frames* aplicados à rede classificatória de 270 para 20 (92,6% de redução). A robustez do modelo é investigada nos datasets FaceForensics++ (subconjuntos C23 e C40) e DFDC, por meio de testes cruzados e análise dos histogramas de predição.

II. A ARQUITETURA F3-NET

A arquitetura F³-Net, proposta por [5] e ilustrada na Figura 1, estabelece um paradigma de análise espectral aprendível. O modelo substitui bancos de filtros estáticos por mecanismos parametrizados que identificam autonomamente os componentes espectrais mais discriminativos, operando em duas vias complementares: a Decomposição Sensível à Frequência (FAD) e a extração de Estatísticas Locais de Frequência (LFS).

A. Decomposição Sensível à Frequência (FAD)

A via FAD particiona adaptativamente o sinal no domínio da frequência. O método emprega $N = 3$ máscaras binárias base no domínio espectral, $\{f_{base}^i\}_{i=1}^N$, acoplados a filtros aprendíveis $\{f_w^i\}_{i=1}^N$. Para conferir estabilidade, esses parâmetros são limitados por uma função de ativação sigmoideal modificada $\sigma(w) = \frac{1 - \exp(-w)}{1 + \exp(-w)}$.

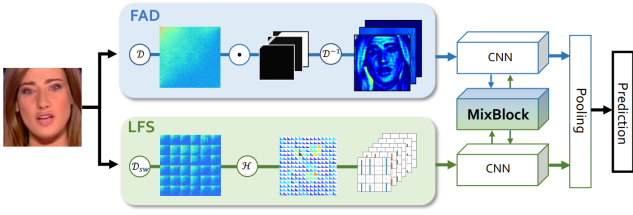


Fig. 1: Visão geral da arquitetura F³-Net segundo [5].

A filtragem ocorre via produto de Hadamard ($\odot$) entre o espectro da imagem e a máscara adaptativa, com retorno ao domínio espacial pela Transformada Cosseno Discreta Inversa (IDCT):

$$y_i = D^{-1} \{ D(x) \odot [f_{base}^i + \sigma(f_w^i)] \}, \quad (1)$$

onde x denota o *frame* de entrada, enquanto D e D^{-1} representam, respectivamente, a DCT bidimensional e a sua inversa, conforme ilustrado na Figura 1.

A formulação geométrica dessas máscaras base fundamenta-se na equalização da energia espectral. Em imagens naturais, a magnitude dos coeficientes da DCT decai exponencialmente em direção às bordas. Para garantir um condicionamento numérico adequado, o espectro é particionado em bandas de baixa (primeiros 1/16 da área), média (do intervalo de 1/16 a 1/8) e alta frequência (os 7/8 restantes).

B. Estatísticas Locais de Frequência (LFS)

Embora a via FAD isole frequências manipuladas, o retorno ao domínio espacial via IDCT faz com que as magnitudes espectrais não sejam aproveitadas como descritores discriminativos. Por outro lado, aplicar a DCT globalmente comprometeria a topologia espacial da imagem exigida pelas operações da CNN [5]. Para solucionar esse conflito, o módulo LFS emprega a DCT em janelas deslizantes, sem realizar a transformada inversa, gerando uma representação híbrida que extrai descritores de frequência locais enquanto preserva a organização estrutural da face. Para cada janela local p , a estatística escalar q_i da banda i (segmentada em $M = 6$ bandas otimizáveis) é extraída por:

$$q_i = \log_{10} \| D(p) \odot [h_{base}^i + \sigma(h_w^i)] \|_1, \quad (2)$$

onde h_{base}^i denota a máscara de filtragem base e h_w^i representa o filtro adaptativo correspondente à banda i , de forma análoga à formulação da via FAD, sendo $\| \cdot \|_1$ a norma L_1 . A escala logarítmica ($\log_{10}$) atua como uma função de estabilização não-linear, vital para equalizar o balanço energético e evitar que as magnitudes das baixas frequências saturam o treino da rede.

O algoritmo opera com janelas de dimensão 10×10 e passo de deslocamento igual a 2. Ao receber uma matriz com resolução espacial de 299×299 , a varredura consolida um volume numérico reduzido de $149 \times 149 \times 6$.

A importância dessa projeção reside no fato de que esse novo tensor atuar como um verdadeiro mapa de calor espectral. Cada pixel não codifica mais luminosidade, mas sim a

energia estatística espectral de uma região específica do rosto, denunciando descontinuidades diretas da manipulação (como falhas de mesclagem nas bordas de uma *deepfake*).

C. Fusão por Atenção Cruzada (MixBlock)

Em arquiteturas multimodais, a integração de múltiplas vias ocorre tipicamente por concatenação linear em profundidade. Contudo, essa abordagem subutiliza a correlação intrínseca entre os domínios. Para superar essa limitação, a arquitetura emprega o *MixBlock*, um mecanismo bidirecional de atenção cruzada onde os mapas de ativação de um domínio operam como tensores de calibração dinâmica para o domínio complementar.

Os fluxos vetoriais, X_F (oriundo do FAD) e X_L (oriundo do LFS), são entrada de duas CNNs independentes e que possuem uma camada de troca de informações via *Transformer* após a última camada convolucional da rede. Essa adaptação exclui um segundo *Transformer* localizado no sétimo bloco da CNN numa tentativa de redução de complexidade computacional. O *Transformer* aplica convoluções 1×1 para gerar matrizes de Consulta (Q), Chave (K) e Valor (V) [6]. A operação de atenção (Y_F) que enriquece o fluxo espectral global (FAD) com as características estatísticas locais (LFS) e é definida por:

$$Y_F = X_F + \text{Softmax} \left(\frac{Q_F \cdot (K_L)^T}{\sqrt{d_k}} \right) \cdot V_L, \quad (3)$$

onde Q_F representa a matriz de consulta do ramo FAD, enquanto K_L e V_L correspondem às chaves e valores do ramo LFS. O termo $\sqrt{d_k}$ atua como fator de escalonamento paramétrico para garantir a estabilidade numérica da função *Softmax*. Análoga e simetricamente a operação de atenção Y_L é calculada, enriquecendo o fluxo LFS com as características FAD.

III. METODOLOGIA E CONFIGURAÇÃO EXPERIMENTAL

O treinamento de arquiteturas de análise espectral demanda intenso fluxo de dados. Para viabilizar a convergência do modelo F³-Net sob restrições de *hardware*, implementou-se um protocolo metodológico focado na otimização da etapa de pré-processamento, tornando-a *offline*.

A. Bases de Dados e Otimização do Pré-processamento

O modelo foi avaliado utilizando os *datasets* Face-Forensics++ (FF++) [3] e *Deepfake Detection Challenge* (DFDC) [7]. O protocolo experimental estabeleceu que ambos os conjuntos fossem utilizados nas etapas de treinamento e avaliação intra-domínio de forma independente. Adicionalmente, realizou-se o teste cruzado, submetendo os modelos treinados em um domínio não visto, visando mensurar a generalização da arquitetura frente a variações de iluminação e métodos de manipulação não vistos durante a etapa de aprendizado.

Em pipelines de treinamento padrão, a decodificação de vídeo e a detecção facial ocorrem em tempo real na CPU, gerando um gargalo que deixa a GPU ociosa à espera de dados. Em observações iniciais deste estudo, o tempo de treinamento completo sob essa abordagem *online* foi de 124

horas. Para eliminar essa latência, adotou-se a separação entre o pré-processamento e o treinamento através de um fluxo de pré-computação *offline*. Com essa estratégia, o tempo efetivo de processamento foi reduzido para 38 horas (1,5 dias), representando uma otimização de aproximadamente 69% no custo temporal e comprovando a necessidade da adaptação para cenários de hardware restrito.

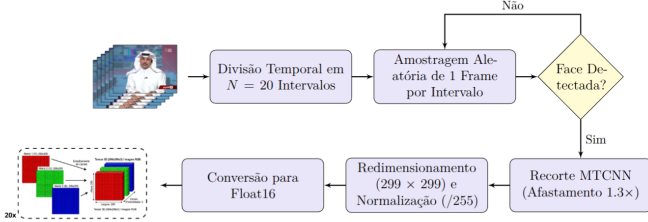


Fig. 2: Fluxograma do pré-processamento, detalhando a amostragem temporal, extração de faces e a adequação dimensional dos tensores pré-computados.

A Figura 2 apresenta a sequência de pré-processamento implementada. O *pipeline* inicia-se com uma amostragem temporal estratificada: cada vídeo foi dividido em $N = 20$ intervalos de tempo de igual duração, e um *frame* foi sorteado aleatoriamente dentro de cada intervalo até que um rosto fosse detectado, num total de 20 *frames* por vídeo. Essa estratégia evita o viés de selecionar apenas *frames* iniciais ou finais e garante cobertura temporal do vídeo. Em seguida, a detecção e o recorte facial foram realizados via MTCNN [8] com fator de escala de $1,3\times$. As faces foram redimensionadas para 299×299 pixels e normalizadas linearmente no intervalo $[0, 1]$. Por fim, os dados foram convertidos para tensores de ponto flutuante de 16 bits (*Float16*) e armazenados em disco, permitindo transferência direta para a VRAM durante o treino e maximizando a eficiência da GPU.

B. Treinamento e Balanceamento de Classes

O conjunto FF++ possui 5 tipos de adulterações que foram combinadas proporcionalmente para compor um total de 5.000 vídeos a serem pré-processados. Para compor o conjunto de vídeos originais, e para corrigir o desbalanceamento de classes, adotou-se a seguinte estratégia para seleção de vídeos: os 1.000 vídeos reais foram divididos em dois grupos de 500: do primeiro, extraíram-se 2 tensores por vídeo (num total de 40 *frames* por vídeo), enquanto que do segundo grupo, foram extraídos 3 tensores por vídeo (i.e., 60 *frames* por vídeo), totalizando 2.500 tensores originais. Em seguida, aplicou-se o espelhamento horizontal nestes tensores, totalizando 5.000 tensores originais. O conjunto DFDC, por sua vez, foi utilizado em sua configuração nativamente balanceada, com 5.000 amostras reais e 5.000 manipuladas.

Os dados foram particionados em 75% (treino), 10% (validação) e 15% (teste). O treinamento ocorreu *end-to-end* via SGD, com *momentum* $\mu = 0,9$, taxa de aprendizado $\eta = 2 \times 10^{-3}$ e decaimento de pesos de 5×10^{-5} . Utilizou-se acumulação de gradientes (4 passos com lote físico de 4 vídeos) para um *batch size* efetivo de 16, ao longo de 20 épocas.

TABELA I: Distribuição da carga paramétrica por blocos funcionais

Módulo Funcional	Parâmetros Treináveis
Filtros Adaptativos FAD	9
Filtros Adaptativos LFS	18
Backbone Xception FAD	22.857.680
Backbone Xception LFS	22.860.272
Fusão por Atenção Cruzada	12.589.056
Classificador Binário Final	2.099.201
Total da Arquitetura	60.406.236

C. Complexidade Paramétrica e Backbone

A arquitetura adota duas instâncias independentes da CNN Xception [9] pré-treinadas na *ImageNet* como *backbone* para os fluxos FAD e LFS (Fig. 1). Os experimentos foram conduzidos no Kaggle utilizando uma GPU NVIDIA Tesla P100 (16 GB VRAM). A Tabela I detalha a distribuição exata do modelo, que totaliza aproximadamente 60,4 milhões de parâmetros treináveis.

D. Estratégia de Inferência e Avaliação

O desempenho do classificador foi mensurado pelas medidas de Acurácia, Precisão, Revocação, *F1-Score* e AUC. O processo preditivo ocorre em duas etapas sequenciais: inferência individual por quadro e agregação temporal por vídeo.

Inicialmente, para cada quadro i dos $N = 20$ processados, a rede gera uma probabilidade de manipulação p_i via ativação Sigmoide. A decisão binária local (f_i) é obtida por:

$$f_i = \text{round}(p_i) = \begin{cases} 1, & \text{se } p_i > 0,5 \\ 0, & \text{caso contrário} \end{cases} \quad (4)$$

A classificação global do vídeo (Y_V) é definida por votação majoritária simples. O vídeo é rotulado como manipulado se as anomalias se repetirem em pelo menos 50% das amostras (limiar de 10 *frames* para $N = 20$).

$$Y_V = \begin{cases} 1, & \text{se } \sum_{i=1}^N f_i \geq \frac{N}{2} \\ 0, & \text{caso contrário} \end{cases} \quad (5)$$

Diferente de ruídos aleatórios de compressão ou oclusões parciais que surgem de forma pontual, as pistas das manipulações tendem a persistir ao longo do vídeo. A exigência de concordância na maioria dos quadros tende a reduzir a ocorrência de falsos positivos no sistema.

IV. RESULTADOS E DISCUSSÕES

Os testes efetuados para avaliar as adaptações propostas ao trabalho de [5] será apresentada, comparativamente, a partir das bases FF++ com leve compressão (C23), com alta compressão (C40) e com a base DFDC.

A Fig. 3 consolida a evolução das curvas de acurácia e perda nas etapas de treino e validação (coluna à esquerda) e a distribuição probabilística das predições (histogramas em % à direita) para os três cenários avaliados. As curvas de convergência de treino e validação mostram que o treinamento foi eficiente e que o treinamento não sofreu *overfitting* ao longo das 20 épocas treinadas.

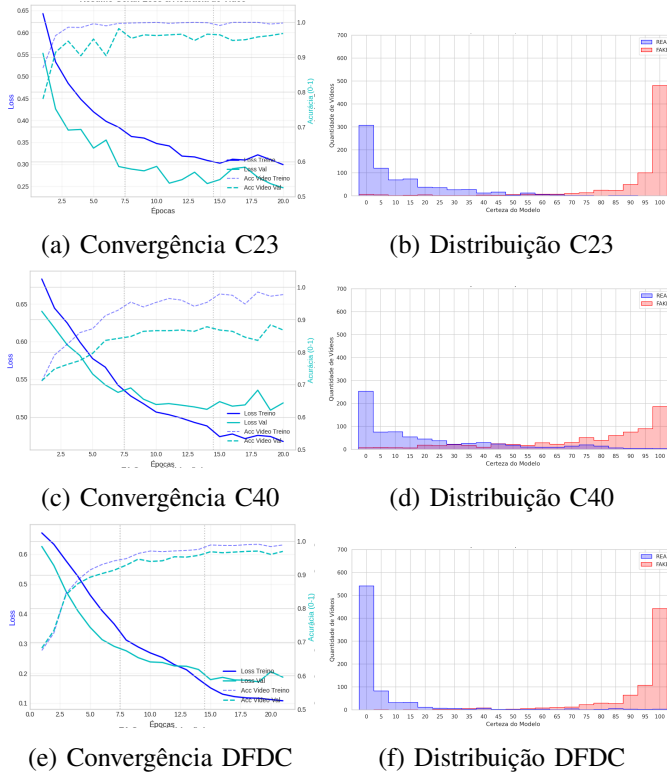


Fig. 3: Desempenho da F³-Net. À esquerda, curvas de treino (azul escuro) e validação (azul claro). À direita, histogramas (%) da incerteza do modelo sob compressão e mudança de domínio.

Os valores apresentados no eixo horizontal das Figuras 3 (b), (d) e (f) foram discretizados em incrementos de 5% (de 0% à 100%). Essa quantização decorre diretamente da estratégia de inferência restrita aos 20 *frames* por vídeo, i.e., que cada *frame* predito como *fake* contribui com exatos 5% para o escore final do vídeo.

Em um cenário analítico ideal, a classificação operaria com bimodalidade absoluta: vídeos originais concentrados em 0% e manipulados em 100%. Embora a bimodalidade não tenha sido atingida, os gráficos demonstram uma expressiva margem de separação na maioria dos casos. As amostras autênticas agrupam-se densamente na faixa de 0% a 30% com um pico em 0%, enquanto as manipuladas concentram-se no extremo oposto, entre 70% e 100% com um pico em 100%. A escassez de densidade no centro dos gráficos comprova que o limiar de decisão de 50% opera em uma zona segura, com reduzida probabilidade de sobreposição de classes.

O cenário mais desafiador ocorre no teste C40 (Figuras 3 (c) e (d)), onde, possivelmente, a severidade da compressão do *codec* H.264, que causa o descarte das assinaturas de alta frequência, provoca uma suavização das pistas de manipulação, causando uma maior incerteza preditiva.

A avaliação no DFDC apresenta a melhor bimodalidade e convergência do estudo, atestando a alta eficácia da rede em mapear e distinguir as manipulações nativas deste *dataset*.

Os resultados numéricos consolidados na Tabela II quantificam o impacto da compressão e da variação de características das bases no desempenho da rede sob a ótica das métricas de

TABELA II: Desempenho comparativo da F³-Net reproduzida aos cenários FF++ C23 e C40 e DFDC.

Cenário	Acc. Vídeo	Acc. Frame	AUC	Precisão	Revocação	F1-Score
FF++ (C23)	96,53%	90,99%	0,9861	0,9820	0,9480	0,9647
FF++ (C40)	84,00%	77,87%	0,9176	0,8612	0,8107	0,8352
DFDC	96,73%	93,40%	0,9924	0,9630	0,9720	0,9675

avaliação escolhidas. As métricas AUC, Precisão, Revocação e F1-Score foram calculadas em nível de vídeo, assim como a acurácia; a acurácia de *frame* é reportada separadamente como referência auxiliar.

No cenário C23, a arquitetura alcançou acurácia de vídeo de 96,53% e AUC de 0,9861, validando a eficácia da fusão espectral em mídias de alta qualidade. Conforme a compressão avança para o nível alto (C40), observa-se uma queda esperada nas métricas (acurácia de 84,00% e AUC de 0,9176).

Por fim, a avaliação no conjunto DFDC registrou o melhor desempenho global do estudo, com acurácia de vídeo de 96,73% e AUC de 0,9924 e obteve um equilíbrio entre a taxa de revocação e a precisão (0,9720 e 0,9630, respectivamente).

A capacidade de generalização da arquitetura foi testada através de avaliações cruzadas, onde o modelo treinado em um *dataset* é submetido a testes em cenários distintos. A Tabela III consolida esses resultados.

TABELA III: Resultados da avaliação cruzada inter-dataset

Cenário (Treino → Teste)	Acc. Vídeo	AUC	F1-Score
C23 → C23 (<i>Baseline</i>)	96,53%	0,9861	0,9647
C23 → C40 (<i>Cruzado</i>)	67,80%	0,8243	0,5645
C23 → DFDC (<i>Cruzado</i>)	62,93%	0,7170	0,5190
C40 → C40 (<i>Baseline</i>)	84,00%	0,9176	0,8352
C40 → C23 (<i>Cruzado</i>)	78,13%	0,8755	0,7461
C40 → DFDC (<i>Cruzado</i>)	61,67%	0,6694	0,5497
DFDC → DFDC (<i>Baseline</i>)	96,73%	0,9924	0,9675
DFDC → C23 (<i>Cruzado</i>)	58,87%	0,6279	0,5873
DFDC → C40 (<i>Cruzado</i>)	55,40%	0,5720	0,5477

Os dados revelam uma degradação de desempenho ao realizar a transição entre *datasets*. O modelo treinado em C23 apresenta dificuldade ao lidar com a compressão C40 e com o DFDC, com quedas acentuadas das três métricas apresentadas. Esse comportamento indica que artefatos espectrais aprendidos em alta qualidade são sensíveis à quantização e à mudança nos métodos de manipulações.

Notavelmente, o modelo treinado em C40 demonstrou maior resiliência ao testar em C23 do que o inverso, sugerindo que o treinamento sob forte compressão leva a rede a aprender características mais robustas. Por fim, o modelo DFDC, apesar de muito eficiente intra-*dataset*, apresentou a menor capacidade de generalização para o FF++, indicando um alto grau de especialização nas manipulações específicas de sua base original.

A Figura 4 apresenta as projeções t-SNE dos vetores de características aprendidas pela Xception no cenário de treinamento e teste intra-classe. Nos cenários C23 e DFDC, observa-se uma separação espacial clara, com o DFDC apresentando os agrupamentos mais densos, especialmente para a classe *fake*. Em contrapartida, o cenário C40 exibe uma sobreposição

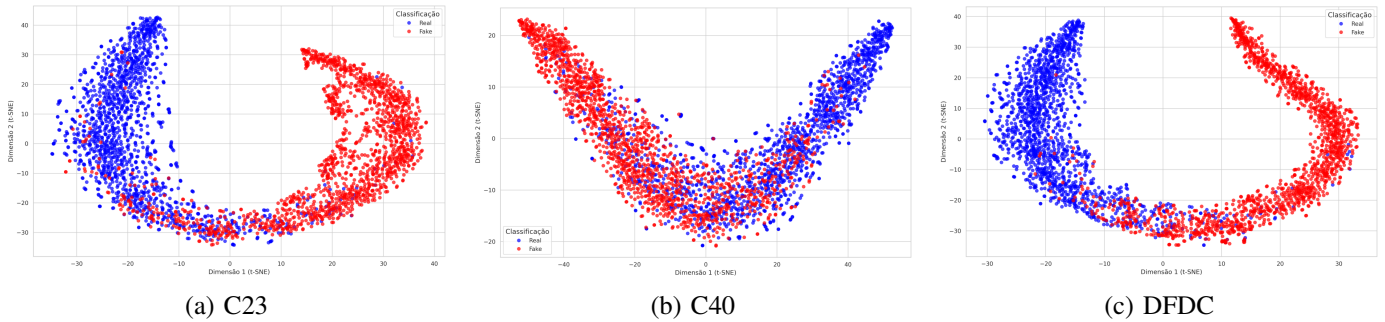


Fig. 4: Visualização t-SNE do espaço latente onde pontos vermelhos indicam *frames* falsos e azul *frames* verdadeiros: (a) dispersão e sobreposição no C23; (b) dispersão e sobreposição maiores em C40; (c) agrupamentos mais densos no DFDC.

entre amostras autênticas e manipuladas, ilustrando como a compressão agressiva suaviza os artefatos de manipulação mesmo dentro do próprio domínio de treinamento.

Para atestar a validade da implementação, a Tabela IV apresenta um comparativo entre o desempenho da arquitetura adaptada e as métricas reportadas na literatura para os níveis de compressão C23 e C40 do *dataset* FaceForensics++, além do conjunto DFDC. A comparação é feita sob o limiar de decisão de $\theta = 0,5$ que define a probabilidade mínima de classificar um vídeo como manipulado (50%).

No cenário C23, os resultados mostram um alinhamento com a proposta original [5], com uma diferença de apenas 0,78%, superando modelos espaciais consolidados como a Xception e a MesoNet. No cenário de compressão alta (C40), a acurácia de 84,00% mantém a consistência com a literatura, apesar da queda esperada.

Ressalta-se que enquanto o trabalho de [5] utiliza 270 *frames* por vídeo, este estudo opera com uma entrada a apenas 20 *frames*, demonstrando alta eficácia mesmo sob restrições de amostragem e processamento. Além disso, em [5], o módulo de atenção cruzada é composto por dois ramos, enquanto a abordagem proposta emprega apenas um.

TABELA IV: Resultados comparativos de acurácia de vídeo para $\theta = 0,5$.

Método	Acc (C40)	Acc (C23)	Acc (DFDC)
<i>Steg.Features</i> [9]	55,98%	70,97%	-
<i>MesoNet</i> [2]	70,47%	83,10%	-
<i>Xception</i> [10]	82,71%	95,04%	-
F ³ -Net (<i>Xception</i>) [5]	86,89%	97,31%	-
F ³ -Net (presente proposta)	84,00%	96,53%	96,73%

V. CONSIDERAÇÕES FINAIS

Este trabalho reproduziu e adaptou o fluxo de processamento da F³-Net para detectar *deepfakes* em *hardware* limitado. A estratégia foi dividida entre tornar a etapa de pré-processamento *offline*, reduzindo gargalos de processamento na CPU, e utilizar apenas 20 *frames* durante a etapa de treinamento da rede. Os testes em múltiplos domínios confirmaram a robustez do modelo, mesmo em vídeos com compressão alta (C40) ou na base DFDC.

Os testes inter-*dataset* mostraram que o treinamento com a base DFDC foi ligeiramente melhor que o nível de acaso

(50%), indicando que as pistas de adulteração constantes nessa base não são representativas no domínio da FF++.

Como próximos passos, pretende-se avaliar o desempenho de *backbones* mais leves, incluindo sua capacidade de generalização para vídeos inéditos e em cenários inter-dataset. Além disso, pretende-se realizar estudos de ablação para consolidar e quantificar a contribuição de cada uma das modificações propostas.

REFERÊNCIAS

- [1] David M. J. Lazer, Matthew A. Baum, Yochai Benkler, Adam J. Berinsky, Kelly M. Greenhill, Filippo Menczer, Miriam J. Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, Michael Schudson, Steven A. Sloman, Cass R. Sunstein, Emily A. Thorson, Duncan J. Watts, and Jonathan L. Zittrain, “The science of fake news,” *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.
- [2] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, “Mesonet: a compact facial video forgery detection network,” *CoRR*, vol. abs/1809.00888, 2018.
- [3] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner, “Faceforensics++: Learning to detect manipulated facial images,” *CoRR*, vol. abs/1901.08971, 2019.
- [4] Ahmad Raza, Abdul Basit, Asjad Amin, Zeeshan Ahmad Arfeen, Muhammad I. Masud, Umar Fayyaz, and Touqeer Ahmed Jumani, “A comprehensive review of deepfake detection techniques: From traditional machine learning to advanced deep learning architectures,” *AI*, vol. 7, no. 2, 2026.
- [5] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao, “Thinking in frequency: Face forgery detection by mining frequency-aware clues,” *CoRR*, vol. abs/2007.09355, 2020.
- [6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, “Attention is all you need,” 2023.
- [7] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton-Ferrer, “The deepfake detection challenge dataset,” *CoRR*, vol. abs/2006.07397, 2020.
- [8] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.
- [9] Jessica Fridrich and Jan Kodovsky, “Rich models for steganalysis of digital images,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.
- [10] François Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1800–1807.