Algoritmo para Detecção de Imagens de Faces Deep Fakes

M. A. C. dos Santos
Universidade Federal de São Paulo - UNIFESP
Instituto de Ciência e Tecnologia
São José dos Campos, São Paulo, Brasil
macsantos23@unifesp.br

R. D. R. de Moraes
Universidade Federal de São Paulo - UNIFESP
Instituto de Ciência e Tecnologia
São José dos Campos, São Paulo, Brasil
raphael.damasceno@unifesp.br

***Resumo*—Diante a problemática do uso incorreto das tecnologias de sintetização de imagens, este projeto tem como objetivo, criar um algoritmo capaz de classificar imagens de faces humanas manipuladas e reais, para isto modelos como de Redes Neurais Artificiais, Deep Learning e métodos como transferência de aprendizagem foram aplicados com objetivo de descobrir qual modelo obterá melhor desempenho em tal tarefa. Foram aplicadas etapas de pré-processamento nos dados, afim de aumentar a acurácia do mesmo e sua generalização diante objetos desconhecidos. Para avaliação dos resultados serão utilizadas métricas como F1-Score que permitirão afirmar qual o melhor modelo, esperando obter um algoritmo satisfátorio para tal tarefa de classificação. E como meio de validação do melhor resultado, um outro conjunto de dados foi utilizado para verificar o desempenho do algoritmo em objetos de diferentes domínios.**

***Index Terms*—Deep Fake, Transfer Learning, Data Augmentation**

I. Introdução

Com o avanço das tecnologias, a sociedade tem enfrentado questões filosóficas relacionadas ao uso adequado das mesmas, em relação aos limites de sua boa utilização, principalmente quando se trata da aprendizagem de máquina e suas aplicações em processamento de imagens. Infelizmente, temos observado um aumento nos crimes cibernéticos que envolvem o uso indevido de algoritmos que sintetizam imagens de faces humanas com alta precisão, conhecidos como *Deepfakes*, como podemos ver na Figura 1. Essas produções que se baseiam em criações de imagens totalmente artificiais e manipulação de imagens, por meio de algoritmos de *Deep Learning* ou (*Generative Adversarial Networks* ou GAN's) [1], que podem facilmente enganar o olho humano com tamanha qualidade, trazendo riscos para a sociedade.

Fig. 1: Aplicações de Deepfakes em imagens de celebridades como Tom Cruise. Fonte: [2].

Ao analisar as últimas notícias em relação à golpes podemos notar a crescente aparição dos que utilizam essas tecnologias para esse fim, em uma reportagem do portal R7 podemos encontrar manchetes como *"'Era muito real': novo golpe utiliza deepfake para imitar imagens e voz de pessoas nas redes sociais"* e *Congresso dos EUA pede medidas mais severas contra "deepfakes"*, o que mostra que meios para prevenção a tais problemas sejam extremamente necessários no momento [3]. Tais aplicações tem trazido preocupações a orgãos internacionais como o *International Risk Governance Center*, que trata em algumas de suas publicações sobre os riscos que tais tecnologias trazem aos governos mundiais [4] .

Este estudo teve como objetivo desenvolver um algoritmo capaz de reconhecer os retratos criados com o uso de *Deepfakes*, a fim de prevenir o uso indevido dessa tecnologia e garantir a segurança da sociedade contra fraudes financeiras e fins indevidos em material pornográfico [5], sendo este em específico o que em 2019 representava 96% do uso de tais produções e outros tipos de crimes cibernéticos [6].

Para a realização do trabalho será utilizado um conjunto de imagens da plataforma *Kaggle* [7], o qual é composto por 140.000 fotos de faces humanas divididas em reais (*real*) e falsas (*fakes*). Tal conjunto se demonstra muito interessante para o trabalho a ser realizado, devido a qualidade das produções artificiais Figura 2, que podem ser classificadas como reais ao serem comparadas com imagens verdadeiras.

Fig. 2: Imagens de faces falsas produzidas por GAN's. Fonte: [7].

Diante dos vários trabalhos existentes que abordam o tema, uma das possíveis abordagens a serem implementadas será a de Transferência de Aprendizagem [8], que será aplicada seguindo os seguintes cenários:

1) Transferência de Aprendizagem sem ajuste fino (*fine tuning*).
2) Transferência de Aprendizagem com ajuste fino (*fine tuning*).
3) Transferência de Aprendizagem sem ajuste fino (*fine tuning*) e com Aumentos de dados.
4) Transferência de Aprendizagem com ajuste fino (*fine tuning*) e com Aumentos de dados.

Considerando que para a aplicação do aumento de dados, diferentes ferramentas serão testadas e implementadas, todas seguindo o mesmo protocolo experimental apresentado acima.

II. Fundamentação Teórica

No campo acadêmico encontram-se diversos trabalhos relacionados acerca do tema de detecção de *Deep Fakes*, na grnade maioria dos trabalhos encontramos soluções aplicadas a famosos conjuntos de dados que em sua maioria se constituem de *Deepfake* de celebridades, fazendo então uso de transferência de aprendizagem com redes como o ViT [9] e desta forma obtendo interessantes resultados. Além de trabalhos que puderam testar aplicações como extração de ruídos das imagens em busca de obter mais informações relevantes, podendo conlcuir que a utilização das imagens originais sem tais extrações proporcionam resultados mais relevantes para a tarefa a ser realizada [10].

Assim como diversos outros trabalhos com o objetivo de fornecer métodos eficazes para o reconhecimento de *Deepfakes*, por meio de trasferência de aprendizagem [11] [12] [13] [14] [15] [16]; neste trabalho serão aplicadas técnicas que possibilitem o modelo a ser treinado e aplicado em um conjunto de dados de imagens do mesmo domínio, que possam possibilitar a maior generalização do modelo a ponto de ser eficaz em imagens pertencentes a outros domínios.

Existem diversos trabalhos que têm se dedicado ao desenvolvimento de métodos eficazes para o reconhecimento de Deep Fakes. Um estudo realizado por Ahmed et al. [11] analisou e apresentou uma pesquisa sobre a detecção e reconhecimento de Deep Fakes utilizando redes neurais convolucionais. Eles investigaram o uso de diferentes arquiteturas de rede, incluindo a VGG-16, e discutiram as abordagens mais promissoras nesse campo.

Cavigelli et al. [12] propuseram um método de detecção de Deep Fakes utilizando redes neurais convolucionais quantizadas com precisão mista. Eles demonstraram que essa abordagem pode alcançar um bom desempenho na detecção de Deep Fakes, ao mesmo tempo em que reduz o custo computacional em relação a modelos de alta precisão.

Neves et al. [13] investigaram sistemas de detecção de Deep Fakes e avaliaram sua capacidade de identificar Deep Fakes em comparação com imagens reais. Eles realizaram um estudo abrangente sobre os métodos de síntese de faces utilizados na criação de Deep Fakes, expondo as vulnerabilidades desses métodos e destacando as limitações dos sistemas existentes.

Li e Lyu [14] abordaram o problema da detecção de Deep Fakes por meio da identificação de artefatos de deformação facial. Eles propuseram uma abordagem baseada em características visuais e espaciais para revelar os sinais de manipulação digital nas faces sintetizadas.

Mirsky e Lee [15] realizaram uma ampla revisão sobre a criação e detecção de Deep Fakes. Eles abordaram as técnicas utilizadas para a criação de Deep Fakes, bem como as estratégias de detecção empregadas para identificar esses vídeos falsificados. O estudo oferece uma visão geral das abordagens existentes e das questões de segurança relacionadas.

Almars [16] realizou uma pesquisa abrangente sobre as técnicas de detecção de Deep Fakes utilizando aprendizado profundo. O autor revisou diferentes abordagens e algoritmos utilizados nessa área, destacando seus pontos fortes e limitações. O estudo fornece um panorama detalhado das técnicas mais recentes e eficazes para detectar Deep Fakes.

Este trabalho em questão se fundamenta nas teorias de aprendizagem de máquina em seu amplo aspecto, de modo que serão aplicados modelos de classificadores como Redes Neurais Artificiais (RNA's), Aprendizagem Profunda (*Deep Learning*), Transferência de Aprendizagem (*Transfer Learning*) [8] e métodos auxiliares como o Aumento de Dados (*Data Augmentation*). Nas subseções a seguir o embasamento teórico dos modelos e explicações sobre seus funcionamentos.

A. Aprendizagem Supervisionada

Dado que os valores de rótulos do conjunto de dados do modelo são conhecidos, pode-se descrever tal rotina de aprendizagem como supervisionada.

Seja um conjunto de entrada com n elementos, pode-se representar como $X = \{x_1, ..., x_n\}$ e um conjunto de saída $Y = \{y_1, ..., y_n\}$, nos quais X e Y representam respectivamente o conjunto de elementos e de rótulos. No modelo de aprendizagem supervisionada a ser utilizado no trabalho os valores de y_i assumem valores discretos $(0, 1)$ tal que 0 representam as imagens reais e 1 as imagens *Deep Fakes*, nomeia-se o modelo de classificação.

B. Aprendizagem Profunda (Deep Learning)

O funcionamento do Deep Learning é baseado em redes neurais artificiais profundas, que consistem em múltiplas camadas de unidades interconectadas, chamadas de neurônios. Cada camada processa e extrai características cada vez mais complexas dos dados de entrada, permitindo a aprendizagem de representações hierárquicas [17] [1].

No contexto do reconhecimento de imagens, o Deep Learning é capaz de aprender automaticamente a detectar e classificar objetos, pessoas e padrões complexos a partir de um grande conjunto de exemplos de treinamento. Isso é alcançado através do treinamento supervisionado, em que as redes neurais são alimentadas com dados rotulados, permitindo que elas aprendam a associar características específicas às suas respectivas classes [17] [18].

Um dos principais benefícios do Deep Learning é a capacidade de realizar aprendizado de características, o que significa que as próprias redes neurais são capazes de aprender quais características são relevantes para a tarefa em questão. Isso

reduz a necessidade de um pré-processamento manual extenso dos dados, permitindo que a rede descubra padrões complexos e não triviais por conta própria [17] [19].

Além disso, o Deep Learning é conhecido por sua capacidade de lidar com dados de alta dimensionalidade e não estruturados, como imagens, texto e áudio. Isso o torna especialmente eficaz em tarefas como reconhecimento de fala, tradução automática, análise de sentimentos e diagnóstico médico [20].

No entanto, o treinamento de modelos de Deep Learning pode ser computacionalmente intensivo e requer grandes conjuntos de dados de treinamento anotados. Além disso, a interpretação dos resultados obtidos por redes neurais profundas pode ser desafiadora, devido à sua natureza complexa e à falta de transparência em relação às decisões tomadas internamente pelos modelos [21].

Apesar desses desafios, o Deep Learning tem se mostrado uma abordagem poderosa para resolver problemas complexos de aprendizado de máquina, impulsionando avanços significativos em áreas como visão computacional, processamento de linguagem natural e robótica autônoma [17].

C. Tranferência de Aprendizagem (Transfer Learning)

A transferência de aprendizado (TL) é um aprendizado que envolve a utilização de conhecimento adquirido em um domínio para melhorar o desempenho em outro domínio relacionado. Esse conceito é baseado na definição de domínios e tarefas, onde um domínio consiste em um espaço de características e uma distribuição de probabilidade marginal, e uma tarefa consiste em um conjunto de rótulos e uma função objetivo [8].

Um problema de transferência de aprendizado ocorre quando temos um domínio-fonte e um domínio-alvo, cada um com suas respectivas tarefas. O objetivo é melhorar o aprendizado da função no domínio-alvo usando o conhecimento adquirido no domínio-fonte, mesmo que os espaços de características ou as distribuições de probabilidade sejam diferentes entre eles.

Existem três questões principais no desenvolvimento de algoritmos de TL: quando transferir, o que transferir e como transferir. Essas questões dizem respeito às situações em que a transferência de conhecimento é aplicável, que tipo de conhecimento pode ser transferido entre os domínios ou tarefas e como esse conhecimento prévio pode ser utilizado de forma eficiente [8].

Quanto ao ”o que transferir”, destacam-se três tipos de informações que podem ser transferidas: instâncias, características e parâmetros. A transferência de instâncias envolve o uso de algumas instâncias rotuladas do domínio-fonte para construir um modelo no domínio-alvo. A transferência de características visa encontrar representações de características relevantes que reduzam a diferença entre os domínios. Já a transferência de parâmetros envolve a busca por parâmetros comuns entre os modelos construídos nos dois domínios.

Com relação ao ”como transferir”, existem três cenários principais de TL: transferência indutiva de aprendizado (ITL), transferência transdutiva de aprendizado (TTL) e transferência de aprendizado não supervisionada (UTL). O ITL envolve o aprendizado entre tarefas diferentes, podendo ser feito de forma supervisionada ou não supervisionada. O TTL assume igualdade entre as tarefas, mas desigualdade entre os domínios, aproveitando informações extras sobre a relação entre eles. O UTL ocorre quando não há dados rotulados em nenhum dos domínios, e as tarefas comuns são agrupamento e redução de dimensionalidade [22] [23].

Em resumo, a transferência de aprendizado busca aproveitar conhecimento adquirido em um domínio para melhorar o desempenho em outro domínio relacionado. Existem diferentes abordagens e cenários de TL, que visam identificar o conhecimento a ser transferido e desenvolver métodos eficazes para transferir esse conhecimento entre os domínios ou tarefas.

D. Aumento de dados (Data Augmentation)

Uma ferramenta que se demonstra útil para melhorias em modelos de processamento de imagens é o Aumento de Dados [1]. Tal método se consiste em uma etapa de pré-processamento de dados, no qual transformações são aplicadas nos elementos Figura 3.
As principais tranformações aplicadas em imagens são:

- Translação;
- Rotação;
- Zoom;
- Adição de ruídos;
- Cortes;
- Mudança de bilho.

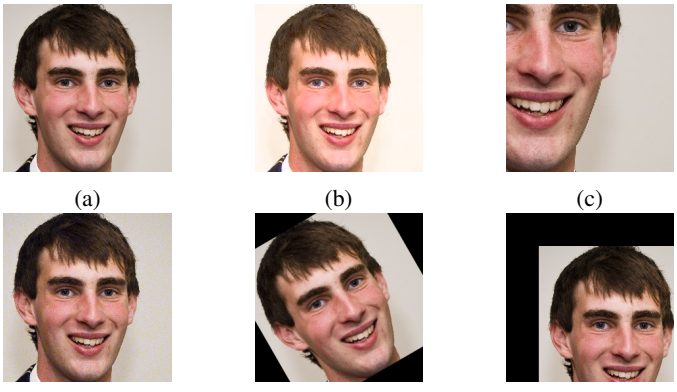

Fig. 3: Aplicação do aumento de dados em uma imagem do conjunto de dados. (a) Figura base; (b) Figura com alteração no brilho; (c) Figura Cortada; (d) Figura com adição de ruído; (e) Figura rotacionada e (f) Figura translacionada. Fonte: [7].

A aplicação desta ferramenta possibilita uma maior generalização do modelo, de modo que o mesmo possa se adaptar a alterações não vistas em treino [24].

1) Aumento de Dados Automático (Auto Augmentation): Um método de aumento de dados a ser aplicado será o *Auto Augmentation* [25], que utiliza um algoritmo de busca para descobrir políticas de aumento de dados que melhoram

o desempenho do modelo. Ele realiza uma pesquisa em um espaço de políticas pré-definidas, combinando diferentes transformações de forma otimizada. O objetivo é encontrar um conjunto de políticas que maximize o desempenho do modelo em um conjunto de validação. Ao automatizar esse processo, o AutoAugment permite que modelos se beneficiem de um aumento de dados mais eficaz, sem a necessidade de ajustes manuais extensivos [26], Figura 4.

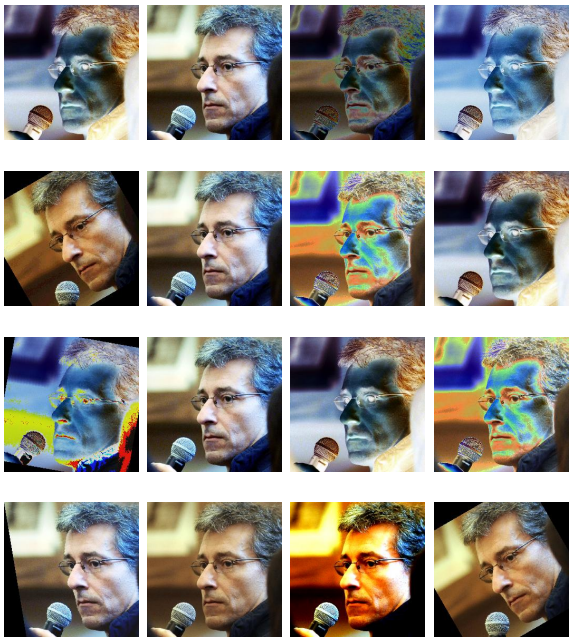

Fig. 4: Imagens com o método de aumento de dados AutoAugment aplicado. Fonte: [7].

2) Aumento Aleatório (Rand Augmentation): Outro método a ser aplicado é o *Rand Augmentation*, que em vez de realizar uma busca por políticas específicas, o RandAugment aplica uma sequência aleatória de transformações em cada imagem durante o treinamento. Essas transformações são selecionadas aleatoriamente de um conjunto pré-definido de operações, como ajuste de brilho, rotação, zoom, entre outros, Figura 5, [27].

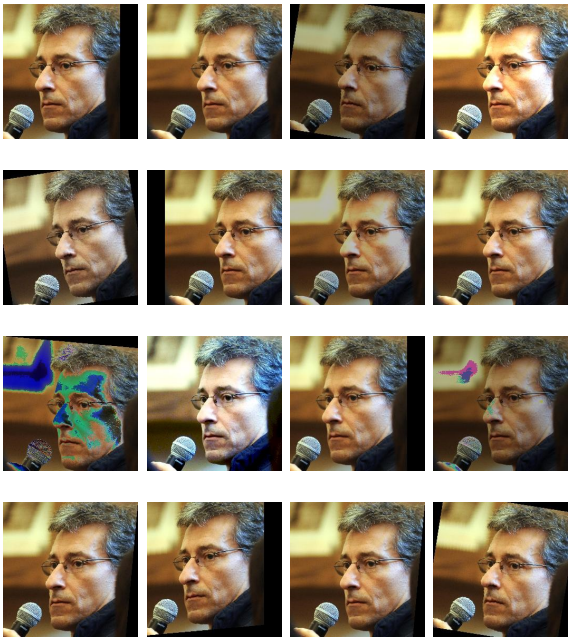

Fig. 5: Imagens com o método de aumento de dados RandAugment aplicado. Fonte: [7].

E. Ferramentas Utilizadas

Para esse projeto, planeja-se a utilização de bibliotecas da linguagem Python, para processamento de imagens, aplicação dos modelos, pré-processamento dos dados e avaliação de desempenho dos mesmos. Em termos de processamento de imagens e para aplicação dos modelos será utilizada a biblioteca PyTorch para os modelos de redes neurais e redes neurais profundas [28]; e para pré-processamento dos dados e avaliação do desempenho seão utilizadas as bibliotecas SciKit-Learn e PyTorch.

F. Modelos de Redes Neurais Utilizados

As redes ConvNext [29] e ViT [30] desempenham um papel fundamental neste projeto, fornecendo abordagens poderosas para a tarefa de processamento de imagens e classificação. A rede ConvNext, baseada em convoluções, é amplamente utilizada em visão computacional e é conhecida por sua capacidade de extrair características relevantes de imagens. Já a rede ViT, baseada em transformers, revolucionou o campo de processamento de imagens ao aplicar a atenção e mecanismos de transformer em vez de convoluções tradicionais. A rede ConvNext será utilizada para realizar o processamento de imagens e extração de características relevantes. Seu design arquitetural, composto por camadas convolucionais em cascata, permite a detecção de padrões e a criação de representações visuais de alta qualidade [29].

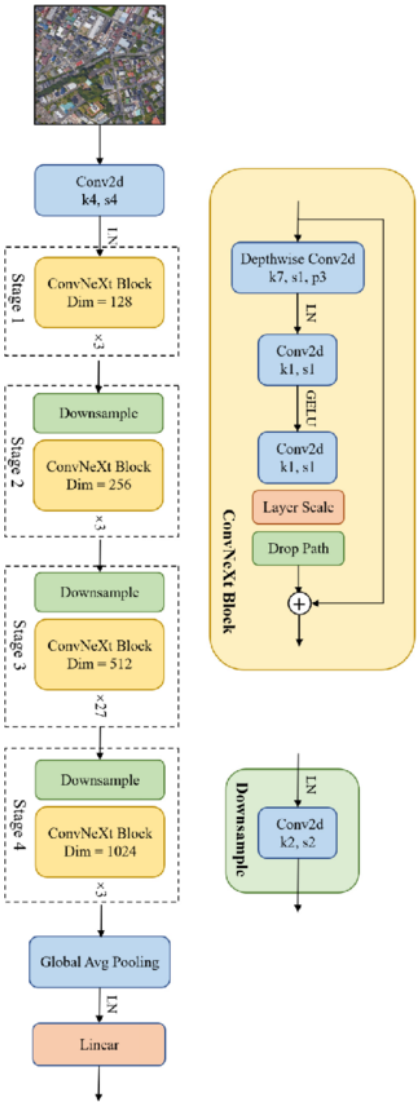

Fig. 6: Arquitetura da Rede ConvNeXt [31]

Por outro lado, a rede ViT será empregada para realizar a classificação de imagens, Figura 10. Ao utilizar mecanismos de atenção e transformers, o ViT é capaz de capturar relações espaciais e contextuais entre os pixels de uma imagem, permitindo uma compreensão mais aprofundada da mesma [30].

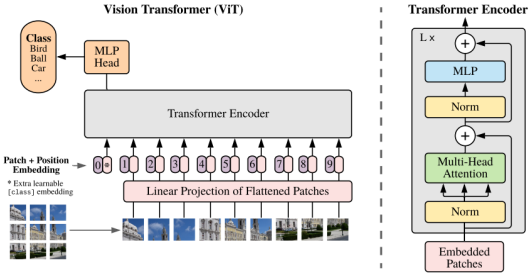

Fig. 7: Arquitetura da Rede ViT [30]

A combinação das redes ConvNext e ViT neste projeto visa explorar as vantagens de ambos os modelos. A rede ConvNext será responsável por extrair características de alto nível das imagens, enquanto o ViT será responsável por realizar a classificação com base nessas características.

III. Metodologia

Esta seção descreve a base de dados a ser utilizada no projeto, o pré-processamento das mesmas e como poderão ser aplicados os modelos de classificação para a realização de tal tarefa.

A. Base de Dados

A base de dados utilizada no projeto se trata de um conjunto com 140.000 imagens separadas por conjuntos de treino, teste e validação Tabela I, além da rotulação das classes que são duas: Reais Figura 9 e Falsas Figura 8.

Tabela I: Distribuição do conjunto de dados

Tipo de Conjunto	Rótulo	Quantidade
Treino	Reais	50.000
	Falsas	50.000
Teste	Reais	10.000
	Falsas	10.000
Validação	Reais	10.000
	Falsas	10.000
Totais		140.000

Fig. 8: Imagens de faces falsas produzidas por GAN's. Fonte: [7].

Fig. 9: Imagens de faces reais capturadas do Flickr. Fonte: [7].

As imagens falsas deste conjunto de dados foram produzidas por StyleGAN's da Nvidea [32], é uma generative adversarial network (GAN), que aborda o desafio de controlar características específicas nas imagens geradas. Ele opera gerando gradualmente imagens de alta resolução a partir de uma resolução baixa. Os principais componentes do StyleGAN incluem a Rede de Mapeamento [32], Módulos de Estilo (AdaIN) [33], remoção da entrada tradicional [34], variação estocástica [35], mistura de estilos [35], truque de truncamento [32] e ajuste fino [9].

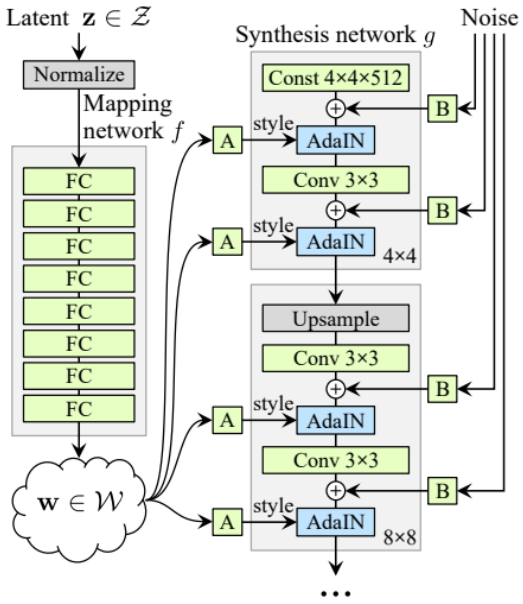

Fig. 10: Arquitetura StyleGAN [32].

B. Protocolo Experimental

O protocolo a ser seguido se consiste no treinamento e teste dos modelos, pré treinados, de modo a avaliá-los nos seguintes cenários em todas suas possíveis permutações:

1) Transferência de Aprendizagem sem ajuste fino (*fine tuning*).
2) Transferência de Aprendizagem com ajuste fino (*fine tuning*).
3) Transferência de Aprendizagem sem ajuste fino (*fine tuning*) e com Aumentos de dados.
4) Transferência de Aprendizagem com ajuste fino (*fine tuning*) e com Aumentos de dados.

Aplicando como métodos de Aumento de dados, AutoAugment [25] e RandAugment [27], teremos um conjunto de cenários finais com 16 possibilidades:

1) ConvNeXt sem ajuste fino (*fine tuning*).
2) ConvNeXt com ajuste fino (*fine tuning*).
3) ConvNeXt sem ajuste fino (*fine tuning*) e com AutoAugment.
4) ConvNeXt com ajuste fino (*fine tuning*) e com AutoAugment.
5) ConvNeXt sem ajuste fino (*fine tuning*) e com RandAugment.
6) ConvNeXt com ajuste fino (*fine tuning*) e com RandAugment.
7) ConvNeXt sem ajuste fino (*fine tuning*) e com AutoAugment e RandAugment.
8) ConvNeXt com ajuste fino (*fine tuning*) e com AutoAugment e RandAugment.
9) ViT sem ajuste fino (*fine tuning*).
10) ViT com ajuste fino (*fine tuning*).
11) ViT sem ajuste fino (*fine tuning*) e com AutoAugment.
12) ViT com ajuste fino (*fine tuning*) e com AutoAugment.
13) ViT sem ajuste fino (*fine tuning*) e com RandAugment.
14) ViT com ajuste fino (*fine tuning*) e com RandAugment.
15) ViT sem ajuste fino (*fine tuning*) e com AutoAugment e RandAugment.
16) ViT com ajuste fino (*fine tuning*) e com AutoAugment e RandAugment.

O dado protocolo pode ser observado pelo diagrama presente na Figura 11, que representa o pipeline de todo processo experimental.

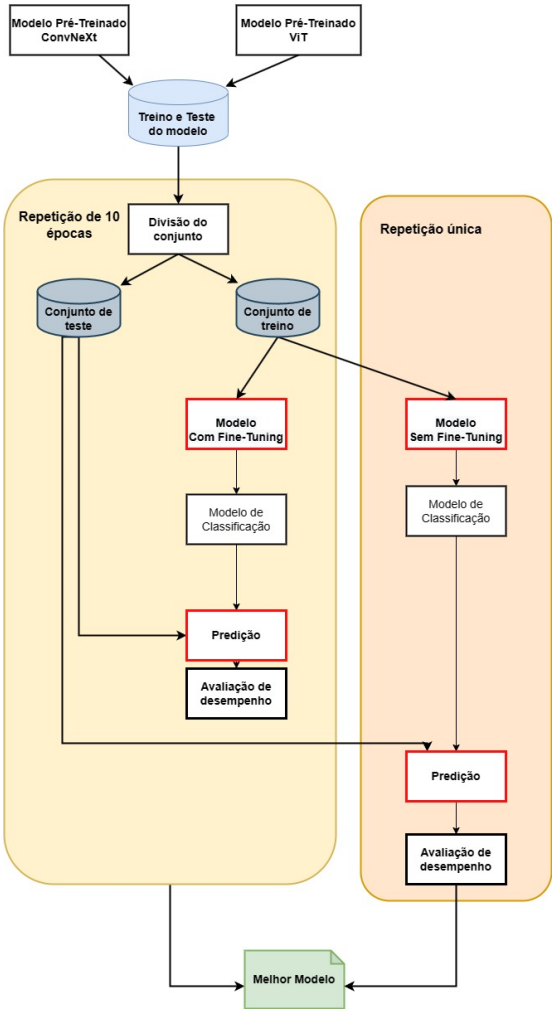

Fig. 11: Pipeline Experimental. Fonte: Próprio

C. Processamento de imagens

Para a etapa de processamento das imagens, foi apenas aplicado um redimensionamento, de modo que as imagens fossem compatíveis com as redes a serem utilizadas, a princípio o tamanho a ser utilizado de forma padrão para as imagens foi o determinado por cada arquitetura a ser utilizada. Para isto utilizou-se a ferramenta de transformação de dados da biblioteca Pytorch da linguagem Python. Nas etapas de aumento de dados, o processo aplicado seguiu o pipeline apresentado na Figura 12, que representa como os diferentes tipos de métodos foram aplicados e utilizados.

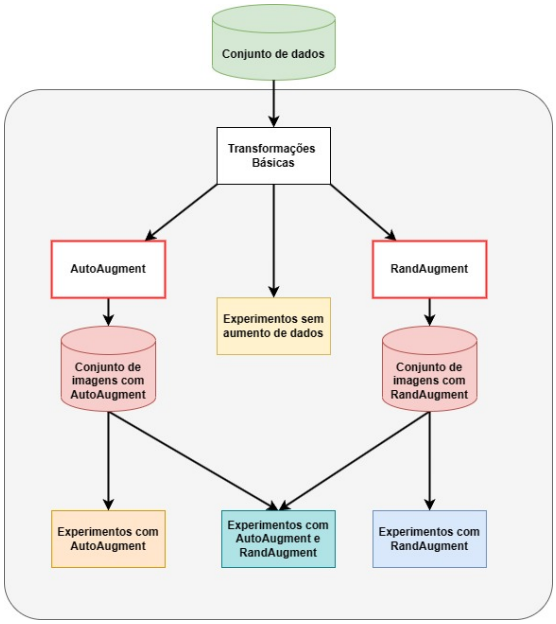

Fig. 12: Pipeline Processamento de Imagens e Aumento de Dados. Fonte: próprio

D. Modelos de Classificação

Diante os modelos citados anteriormente, para cada um deles, foi aplicado o treinamento e teste utilizando o conjunto de dados em busca de avaliar qual modelo possui o melhor desempenho na tarefa de classificação. Para a realização dos treinos e testes das redes foi utilizado um décimo do conjunto total, de modo a otimizar a realização dos testes em relação ao tempo necessário para execução, tendo em vista que a utilização completa do conjunto necessitaria de muitos recursos computacionais e tempo.

E. Tranferência de Aprendizagem

Ao tratar do método de transferência de aprendizagem, aplicaremos conhecidas redes neurais que foram previamente em grandes conjuntos de dados, como as redes ConvNeXt [29] e ViT [30].

F. Métricas de avaliação

Para avaliar os modelos, foram aplicados durantes seus treinamentos, métricas para analisar suas funções de perda e avaliações de desempenho. Diante a tarefa proposta, é possivel afirmar que trata-se de um problema binário, visto que as classes podem ser divididas em apenas duas (0 e 1, sendo respectivamente, o valor da classe falsa e da classe verdadeira), serão utilizadas métricas para avaliar as predições realizadas pelos modelos , sendo estas:

- VP (Verdadeiro Positivo): Quando a predição é da classe positiva e é correta;
- VN (Verdadeiro Negativo): Quando a predição é da classe negativa e é correta;
- FP (Falso Positivo): Quando a predição é da classe positiva e é incorreta;
- FN (Falso Negativo): Quando a predição é da classe negativa e é incorreta.

A métrica de avaliação de desempenho a ser empregada será o F1-Score, que pode ser descrito pela seguinte fórmula:

$$F1\ Score = \frac{(2 \cdot Precision \cdot Recall)}{(Precision + Recall)}$$

Tal que o $Precision$ é a métrica que avalia dentre todas classificações da classe positiva (1) feitas pelo modelo, quantas foram corretas e o $Recall$ a métrica que avalia dentre todos objetos esperados como positivo, quantos foram corretamente classificados, de modo que o $F1 - Score$ representa a média harmônica entre tais [36]. As métricas citadas são definidas pelas seguintes fórmulas:

$$Precision = \frac{VP}{VP+FP}$$

$$Recall = \frac{VP}{VP+FN}$$

Para o trabalho realizado, avaliou-se seu desempenho durante o treinamento e teste, de modo a selecionar os melhores modelos que maximizam o acerto, consequentemente minimizando os erros.

IV. Resultados

A. Resultados Esperados

O principal resultado esperado deste estudo é o desenvolvimento de um algoritmo de Machine Learning capaz de detectar imagens de faces sintetizadas artificialmente, conhecidas como DeepFakes. O algoritmo será treinado com uma combinação de imagens de faces reais e manipuladas, utilizando modelos classificadores de aprendizagem de máquina, como Redes Neurais Artificiais e Deep Learning [1] [17]. Serão aplicados métodos como transferência de aprendizagem e data augmentation para melhorar a eficiência e confiabilidade da classificação [8].

Espera-se que o algoritmo proposto seja capaz de identificar com precisão as imagens de DeepFakes, prevenindo assim o uso indevido dessa tecnologia e garantindo a segurança da sociedade contra fraudes financeiras e uso indevido de material pornográfico [20]. O reconhecimento eficaz de DeepFakes contribuirá para evitar prejuízos financeiros e sociais causados por crimes cibernéticos e abusos dessas aplicações.

Uma vez desenvolvido, o algoritmo poderá ser aplicado em diferentes camadas de validação e pré-processamento de dados, antes de sua utilização final em redes sociais e aplicações bancárias. Isso permitirá identificar qualquer problema relacionado a DeepFakes antes que esses dados sejam publicados, proporcionando uma camada adicional de segurança [14]. Da mesma forma, o algoritmo poderá ser aplicado na detecção de materiais pornográficos criados com o uso indevido de DeepFakes, inibindo sua publicação e disseminação [13].

Em suma, os resultados esperados deste estudo incluem o desenvolvimento de um algoritmo eficiente de detecção de DeepFakes, capaz de prevenir o uso indevido dessa tecnologia e evitar danos financeiros e sociais causados por fraudes e má utilização. Essa contribuição é crucial para abordar

questões críticas relacionadas à segurança cibernética e requer conscientização por parte dos governos, empresas e sociedade em geral sobre os riscos associados ao uso de tecnologias de Deep Learning, bem como a implementação de medidas preventivas para proteger informações pessoais e evitar o uso indevido dessas tecnologias [20].

B. Resultados Obtidos

Ao realizar o protocolo experimental foram obtidos os seguintes resultados a respeito de cada modelo em seus respectivos experimentos, Tabela II e Tabela III.

Tabela II: Resultados - Treinamento (Média de valores)

Modelo	Aumento de dados	Ajuste fino	Loss	Acurácia	F1-Score
ConvNeXt	Sem	Sem	43.21%	53.87%	53.19%
		Com	0.08%	99.44%	99.44%
	AutoAugment	Sem	4.39%	47.51%	47.38%
		Com	0.21%	98.58%	98.58%
	RandAugment	Sem	4.53%	46.20%	38.01%
		Com	0.16%	98.97%	98.97%
	Auto+Rand	Sem	4.32%	53.62%	50.44%
		Com	0.30%	97.95%	97.95%
ViT	Sem	Sem	4.68%	45.93%	44.63%
		Com	0.47%	96.54%	96.54%
	AutoAugment	Sem	4.76%	44.41%	41.14%
		Com	1.35%	90.05%	90.05%
	RandAugment	Sem	4.46%	52.51%	50.95%
		Com	0.85%	93.98%	93.98%
	Auto+Rand	Sem	04.68%	46.94%	46.69%
		Com	2.17%	83.32%	83.32%

Tabela III: Resultados - Teste (Média de valores)

Modelo	Aumento de dados	Ajuste fino	Loss	Acurácia	F1-Score
ConvNeXt	Sem	Sem	8.64%	53.15%	52.62%
		Com	0.12%	99.71%	99.71%
	AutoAugment	Sem	8.76%	47.49%	47.88%
		Com	0.09%	99.76%	99.75%
	RandAugment	Sem	9.12%	44.35%	36.06%
		Com	0.11%	99.64%	99.64%
	Auto+Rand	Sem	8.62%	54.20%	50.68%
		Com	0.13%	99.63%	99.62%
ViT	Sem	Sem	9.26%	47.05%	45.88%
		Com	3.26%	92.36%	92.34%
	AutoAugment	Sem	9.35%	45.95%	42.94%
		Com	9.16%	73.22%	71.55%
	RandAugment	Sem	8.92%	52.85%	50.50%
		Com	3.42%	89.17%	89.07%
	Auto+Rand	Sem	9.69%	44.65%	41.68%
		Com	5.11%	81.55%	81.39%

Ao analisar os resultados obtidos, fica evidente que os experimentos sem ajuste fino no treinamento levam a resultados insatisfatórios. Isso indica que os modelos pré-treinados não são eficazes para resolver esse problema sem a necessidade de ajustes em seus parâmetros. Essa constatação é reforçada pela análise dos resultados dos modelos com ajuste fino durante o treinamento, os quais apresentaram resultados excelentes. Em particular, destaca-se o modelo ConvNeXt com AutoAugment e ajuste fino, que obteve uma média de acurácia nos treinos e testes próxima a 99.88%.

Pode-se também visualizar os resultados gerais de forma gráfica por meio das Figuras 14 e 13, o que demonstra o desempenho dos experimentos com e sem ajuste fine *fine-tuning* e suas variações com aumentos de dados, podendo reforçar a análise da pouca eficácia dos modelos que não possuem ajustes em seus parâmetros.

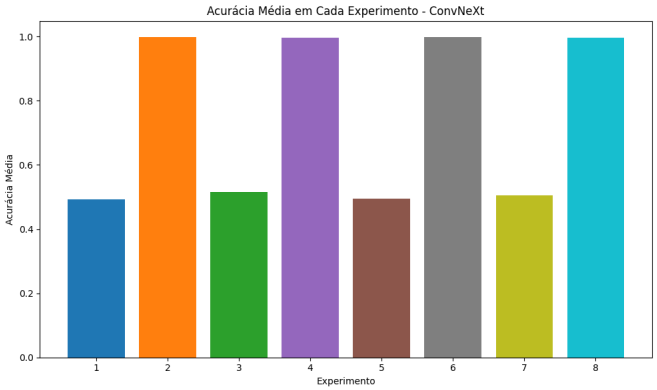

Fig. 13: Gráfico de barras com as médias de acurácia do modelo ConvNeXt em cada um de seus experimentos. Fonte: Próprio.

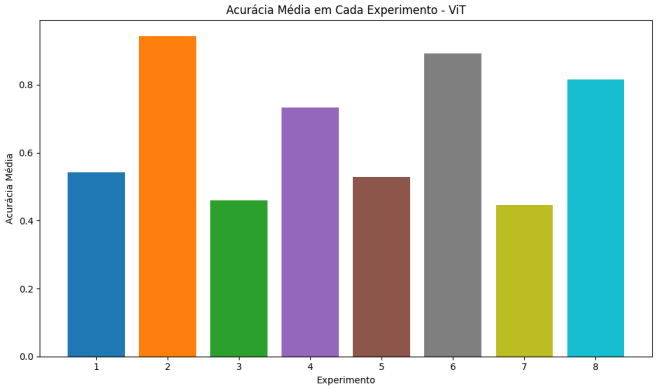

Fig. 14: Gráfico de barras com as médias de acurácia do modelo ViT em cada um de seus experimentos. Fonte: Próprio.

Além disso, observa-se que o melhor modelo ConvNeXt possui uma curva de aprendizagem mais acentuada, Figura 15, em comparação com o melhor modelo ViT, Figura 16, o que sugere que ele requer menos tempo de treinamento para alcançar resultados satisfatórios.

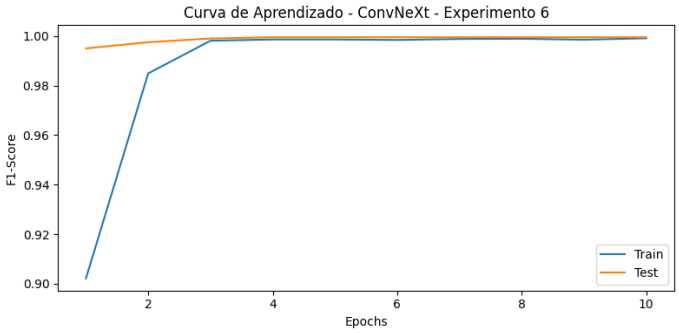

Fig. 15: Curva de aprendizagem do experimento com o modelo ConvNeXt, com melhor resultado obtido. Fonte: Próprio.

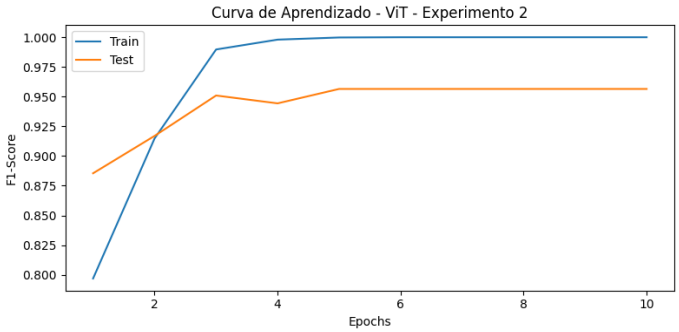

Fig. 16: Curva de aprendizagem do experimento com o modelo ViT, com melhor resultado obtido. Fonte: Próprio.

C. Aplicações dos Resultados

Ao analisar o problema abordado neste projeto de forma algébrica, constatamos que as imagens Deepfakes analisadas são todas geradas a partir do mesmo domínio, usando a abordagem StyleGAN [32]. Portanto, podemos afirmar que nosso modelo é capaz de mapear efetivamente os objetos desse domínio específico. No entanto, surge a questão de como o modelo lidaria com objetos provenientes de diferentes domínios, ou seja, Deepfakes geradas por algoritmos distintos.

Para avaliar o desempenho do modelo em conjuntos de Deepfakes diferentes, será seguido o protocolo a seguir:

1) Coleta do modelo com melhor resultado no conjunto principal;
2) Aplicação do modelo para predição em imagens do novo conjunto:
   a) Aplicação com ajuste fino e Aumento de Dados no novo conjunto.
3) Análise dos resultados.

Tendo em vista a qualidade das imagens geradas pelas StyleGAN's [32] utilizadas, espera-se que o modelo possa apresentar um resultado satisfatório quando submetido a outras produções. Para isto será utilizado um famoso conjunto de imagens Deepfakes, *FaceForensics* [37], que possui diversas imagens de faces manipuladas por Deepfakes que foram extraídas de vídeos, Figura 17.

Fig. 17: Imagens de faces falsas produzidas por diversas GAN's. Fonte: [37].

Tabela IV: Descrição do Conjunto de dados a ser utilizado para validação do modelo com imagens de outros domínios.

Classe	Conjunto Fonte	Quantidade	Total
Reais	140K Real and Fake Faces	10,000	10,000
Falsas	140K Real and Fake Faces	10,000	22,199
	FaceForensics	12,199	

1) Resultados e discussões: Os resultados da aplicação confirmam que o modelo ConvNeXt apresenta excelente funcionamento em tarefas de detecção de Deepfakes, de modo que com pouco treinamento, o modelo já anteriormente treinado com o conjunto *140k Real and Fake faces* [7], foi capaz de identificar com acurácia próxima de 99.80% no conjunto *FaceForensics*. Deste modo podemos concluir que a ferramenta obtida nesta pesquisa possui grandes aplicações para a tarefa desejada, ainda que possa não funcionar de maneira generalizada e sem treinamento para conjuntos de outros domínios, porém com poucos ajustes a mesma passa a ter resultados satisfatórios.

Referências

[1] I. J. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, http://www.deeplearningbook.org.

[2] C. Ume, "Tom Cruise DeepFake," Instagram, 2021. [Online]. Available: https://www.instagram.com/vfxchrisume/?hl=en

[3] R7, "Congresso dos eua pede medidas mais severas contra deepfakes," Disponível em: https://noticias.r7.com/tecnologia-e-ciencia/congresso-dos-eua-pede-medidas-mais-severas-contra-deepfakes-09012020, 2020, acesso em: 03 maio 2023.

[4] A. Collins, "Forged authenticity: governing deepfake risks," EPFL International Risk Governance Center (IRGC), Tech. Rep., 2019.

[5] C. Gosse and J. Burkell, "Politics and porn: how news media characterizes problems presented by deepfakes," *Critical Studies in Media Communication*, vol. 37, no. 5, pp. 497–511, 2020.

[6] TechTudo, "96% dos vídeos de deepfake têm conteúdo pornográfico: veja sete fatos," Online, 2019, disponível em: https://www.techtudo.com.br/listas/2019/10/96percent-dos-videos-de-deepfake-tem-conteudo-pornografico-veja-sete-fatos.ghtml.

[7] xhlulu, "140k real and fake faces," https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces, 2021.

[8] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[9] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *Advances in neural information processing systems*, vol. 33, pp. 12 104–12 114, 2020.

[10] C. Lopes, "Métodos de detecção de imagens deepfake baseadas em modelos generativos," 2022.

[11] S. R. Ahmed, E. Sonuç, M. R. Ahmed, and A. D. Duru, "Analysis survey on deepfake detection and recognition with convolutional neural networks," in *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. IEEE, 2022, pp. 1–7.

[12] L. Cavigelli *et al.*, "Deepfake detection with mixed-precision quantized convolutional neural networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 5, pp. 1742–1756, 2021.

[13] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, V. Lopes, and H. Proença, "Real or fake? spoofing state-of-the-art face synthesis detection systems," *arXiv preprint arXiv:1911.05351*, vol. 2, p. 6, 2019.

[14] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *arXiv preprint arXiv:1811.00656*, 2018.

[15] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 1, pp. 1–41, 2021.

[16] A. M. Almars, "Deepfakes detection techniques using deep learning: a survey," *Journal of Computer and Communications*, vol. 9, no. 5, pp. 20–35, 2021.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[19] A. F. Agarap, *Deep learning using rectified linear units (relu)*, 2018.

[20] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 427–436.

[21] H. Raval and S. Srivastava, "Deepfake detection techniques: A review," *IEEE Access*, vol. 8, pp. 39 707–39 726, 2020.

[22] M. Bahadori, Y. Liu, and D. Zhang, *Learning with Minimum Supervision: A General Framework for Transductive Transfer Learning*, 2011.

[23] R. Caruana, "Multitask learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, 1997.

[24] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27*. Springer, 2018, pp. 270–279.

[25] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation strategies from data," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 113–123.

[26] R. Hataya, J. Zdenek, K. Yoshizoe, and H. Nakayama, "Faster autoaugment: Learning augmentation strategies using backpropagation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*. Springer, 2020, pp. 1–16.

[27] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 702–703.

[28] E. Stevens, L. Antiga, and T. Viehmann, *Deep learning with PyTorch*. Manning Publications, 2020.

[29] J. Qi, M. Nguyen, and W. Q. Yan, "Waste classification from digital images using convnext," in *Image and Video Technology: 10th Pacific-Rim Symposium, PSIVT 2022, Virtual Event, November 12–14, 2022, Proceedings*. Springer, 2023, pp. 1–13.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[31] S. Chen, Y. Ogawa, C. Zhao, and Y. Sekimoto, "Large-scale individual building extraction from open-source satellite imagery via super-resolution-based instance segmentation approach," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 195, pp. 129–152, 2023.

[32] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[33] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.

[34] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[35] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[36] R. Yacouby and D. Axman, "Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models," in *Proceedings of the first workshop on evaluation and comparison of NLP systems*, 2020, pp. 79–91.

[37] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," in *International Conference on Computer Vision (ICCV)*, 2019.